UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

TRABAJO FIN DE GRADO

PROGRAMA DE SOPORTE A LA EXTRACCIÓN DE NOTICIAS Y COMENTARIOS DE LECTORES DE PERIÓDICOS ONLINE, Y ANÁLISIS DE LAS EMOCIONES QUE TRANSMITEN

Jorge de Andrés Nogales

JULIO 2014

Resumen

Este documento describe el trabajo de fin de grado realizado. El objetivo principal de este proyecto es el análisis del impacto emocional de las noticias en los lectores de periódicos online. Esto será útil para analizar una parte de toda la cantidad de datos que se genera cada día en la red y que es imposible procesar de manera manual.

Para ello se ha desarrollado una aplicación capaz de obtener toda la información relativa a las noticias de un periódico online y a los comentarios publicados por los usuarios. Esta información extraída es almacenada para ser usada por una segunda aplicación desarrollada que analiza las emociones vertidas en los textos de las noticias y de los comentarios.

Por último, los datos obtenidos son representados mediante gráficas con las que se podrá estudiar con detenimiento las emociones surgidas tanto de las noticias como de sus comentarios.

Palabras clave

Impacto emocional, extracción de información, análisis emocional, periódico online, extracción de emociones.

Abstract

This document describes the work done for the Final Grade Project. The main objective of this project is the analysis of the emotional impact of the news on online newspapers readers. This will be useful in analyzing a portion of the entire amount of data generated every day which is impossible to process manually.

To do this an application able to get all the information regarding the news from an online newspaper and the comments published by the users has been developed. This obtained information is stored in order to be used by a second developed application which analyzes the expressed emotions from the text of the news and comments.

Finally, the obtained data are represented by charts with which will be able to carefully study the arisen emotions from both the news and its comments.

Keywords

Emotional impact, information extraction, emotional analysis, online newspaper, emotion extraction.

Índice

Índice de tablas

Índice de figuras

Glosario

- HTML (HiperText Markup Language): Lenguaje en el que están escritas las páginas web.
- XML (Extensible Markup Language): Lenguaje basado en etiquetas que sirve para almacenar datos.
- URL (Uniform Resource Locator): Dirección de localización de un recurso, normalmente se trata de direcciones web.
- RSS (Really Simple Syndication): Tecnología basada en XML para distribuir información actualizada.
- NetBeans: Entorno de diversos lenguajes de programación.
- Jsoup: Herramienta de Java que permite descargar, analizar y extraer información de código HTML.
- DOM (Document Object Model): Interfaz de programación de aplicaciones que proporciona un conjunto estándar de objetos para representar HTML y XML.
- XStream: Herramienta de Java que permite almacenar en un fichero con estructura XML objetos de Java.
- JFreeChart: Herramienta de Java que permite realizar diagramas y gráficas a partir de datos.

1.Introducción

1.1.Motivación

La cantidad de datos que se genera día a día en la red es inmensa y esta cantidad va aumentando cada día que pasa. Procesar tal cantidad de información para un ser humano es, de lejos, una meta imposible. Incluso pudiéndose procesar todos estos datos existe información implícita en ellos que no es visible a simple vista.

Una parte de toda esta gran masa de información son los comentarios que genera la gente en diferentes sitios de Internet. Los datos que se almacenan son simplemente palabras, pero la información que las personas transmiten va más allá de eso. Una información que pueden transmitir mediante los comentarios es su estado emocional a través de las palabras que utilizan.

Los periódicos online publican diariamente noticias y éstas pueden provocar en las personas ciertas emociones al leerlas. Conocer lo que sienten las personas sobre un determinado acontecimiento de actualidad puede ser una información muy útil para los periódicos y para la sociedad en general. Sabiendo las emociones o reacciones que provocan ciertas noticias se podrían crear secciones en los periódicos dedicadas a este tipo de noticias, como por ejemplo noticias que produzcan sólo alegría. Es por eso por lo que se ha decidido llevar a cabo este proyecto.

1.2.Objetivo

Este trabajo tiene como objetivo principal analizar el impacto emocional de las noticias presentadas en periódicos online.

Este objetivo general se desglosa en otros objetivos concretos referidos a la extracción de la información de las noticias y comentarios y a su análisis emocional:

- Obtener las noticias de un periódico online.
- Obtener la información de los textos de las noticias.
- Obtener la información de los comentarios de cada noticia.
- Almacenar la información extraída de las noticias y comentarios.
- Recuperar la información extraída previamente de las noticias y comentarios.
- Analizar emocionalmente textos de las noticias y comentarios.
- Representar gráficamente los datos obtenidos de dicho análisis.

1.3. Estructura del documento

A continuación se describe la organización y el contenido del resto de secciones de las que se compone este documento.

Sección 1: Introducción. Explicación de la motivación, los objetivos y la descripción de la estructura del documento.

Sección 2: Estado del arte. Descripción de técnicas e investigaciones relacionadas con en el ámbito de este trabajo.

Sección 3: Análisis. Por un lado se describe el problema y los requisitos necesarios para la resolución del proyecto y por el otro se describen las tecnologías a utilizar.

Sección 4: Diseño. Descripción de la arquitectura general del sistema, la estructura de datos y los módulos de extracción y análisis emocional.

Sección 5: Desarrollo. Explicación de la implementación del diseño anterior.

Sección 6: Ejemplos de análisis de noticias. Ejemplo de ejecución de la aplicación desarrollada y del análisis de varias noticias.

Sección 7: Pruebas y resultados. Descripción de las pruebas realizadas y los resultados obtenidos.

Sección 8: Conclusiones. Exposición de las posibilidades y las limitaciones de la aplicación desarrollada y sus posibles aplicaciones en un trabajo futuro.

2.Estado del arte

A continuación se describirán brevemente una serie de trabajos relacionados con el proyecto.

Comenzando por las investigaciones de extracción de información, en uno de los artículos estudiados [1] se describe el análisis automático de corpus textuales para la creación de diccionarios de patrones de extracción. En otro estudio [2] se desarrolla una herramienta de descarga y análisis de páginas web para la extracción de información. Para ello se realiza un análisis morfo-sintáctico del texto de la página web descargada para poder distinguir cada tipo de información contenida en el código HTML. En el caso de este trabajo, se estudiará el patrón que sigue cada noticia para saber dónde se pueden encontrar los elementos a extraer.

Por otro lado, se han estudiado investigaciones sobre análisis de textos. En el trabajo descrito en [3] se realiza un análisis sintáctico de textos con el fin de asignar etiquetas gramaticales a las palabras, centrándose en la eliminación de ambigüedades. En este trabajo se utilizarán diccionarios de palabras y raíces de palabras a detectar. Los detalles se describirán más adelante.

En el ámbito de análisis de noticias podemos encontrar en el periódico online 20Minutos [4], en la parte dedicada a las noticias más relevantes, un estudio de actividad social sobre cada una de las noticias mostradas. Este estudio, denominado ECO, se realiza utilizando distintos parámetros sobre la actividad relacionada directamente con esa noticia tanto en la comunidad de 20minutos.es como en las redes sociales. 20Minutos ha sido el periódico elegido para la realización de este trabajo, en el que se analizarán los textos de las noticias, los comentarios de los usuarios e información adicional sobre cada uno de ellos.

Por último, se han buscado estudios relacionados con el análisis emocional. En la investigación [5] se realiza un análisis emocional de redacciones realizadas por alumnos en un entorno de e-learning de forma manual. Este análisis se realiza utilizando cuatro diccionarios de palabras. Cada uno contiene palabras que expresan una emoción concreta: Joy (alegría), Anger (ira), Fear (Miedo) y Sadness (Tristeza). Están divididos a su vez en tres tipos de palabras base, posibilitando la detección de palabras similares.

Los autores han facilitado estos diccionarios de emociones para la realización de este proyecto. A partir de ellos se han añadido las terminaciones de las palabras para encontrar variaciones de las mismas que también expresen esa emoción. Utilizando todas estas nuevas palabras se ha realizado el análisis emocional de noticias extraídas de forma automática, comparando cada palabra de los textos con las palabras del diccionario. Con los datos de las emociones detectadas la aplicación genera gráficos para poder analizar de manera detallada los resultados obtenidos.

3.Análisis

Se necesita crear una herramienta que sea capaz de extraer la información de noticias de periódicos online automáticamente y que pueda almacenar dicha información de manera organizada. Además se necesita realizar otra herramienta que lea la información de las noticias anteriormente guardada y analizar las emociones de los textos y comentarios. Esta última herramienta deberá poder representar gráficamente el análisis realizado previamente.

Debido a que cada periódico muestra distinto tipo de información en cada noticia, la forma en que se extraerá la información dependerá del periódico escogido. Por tanto, primero se escogerá un periódico con el que realizar el sistema de extracción. A continuación se presenta el análisis de requisitos realizado.

3.1.Selección de periódico online

La herramienta de análisis emocional está basada en el castellano, por lo que el periódico debe tener este idioma. Como origen de extracción de las noticias y comentarios se han planteado los tres periódicos con más lectores en España.

ElMundo.es cuenta con un sistema de comentarios lineal. Se indica el número de comentarios de cada noticia y se muestran los diez primeros. Para poder ver más comentarios es necesario pulsar en el botón “Ver anteriores”. Existe la posibilidad de ordenar los comentarios por mejor valoración, aunque no se muestra dicho valor.

ElPais.com tiene un sistema de comentarios bajo una herramienta social llamada Eskup. Las noticias de El Pais.com no muestran al inicio los comentarios y no aparecen hasta que se ha llegado al final de la noticia. Los comentarios están diseñados para poder responderse unos a otros, mostrando las respuestas a otros comentarios en modo cascada.

20minutos.es es el tercer medio más leído en España [6]. Cuenta con un sistema de comentarios lineal, mostrando los primeros diez comentarios y un sistema de paginado para navegar por los siguientes. Cada comentario tiene una puntuación, que es la diferencia entre los votos positivos y negativos recibidos por el resto de usuarios. Además muestra un sistema de actividad social, que indica el nivel de actividad de la noticia.

El periódico elegido, sin perjuicio de poder añadir otros más adelante, es 20minutos.es. A diferencia de los otros periódicos se puede obtener la puntuación de cada comentario y esta información puede ser de utilidad en el estudio de la emoción que transmite la noticia. Además, con el sistema de paginado es más fácil extraer la información. El sistema para obtener los siguientes comentarios de ElMundo.es y la herramienta Eskup de ElPaís.com añaden demasiada complejidad a la tarea de extracción, motivo por el cual se han descartado estos periódicos para este trabajo fin de grado.

3.2. Requisitos

Para poder extraer la información de las noticias de un periódico y analizar las emociones transmitidas se han de satisfacer los siguientes requisitos:

3.2.1. Requisitos funcionales:

1. Requisitos para la extracción de información:
    1. Obtención de noticias a analizar: Se debe poder extraer noticias de manera automática de un periódico online con su información asociada.
    2. Obtención de información de la noticia: Se debe poder obtener de cada noticia la siguiente información: URL, titular, texto, autor, fecha y votos recibidos.
    3. Obtención de información de los comentarios: Para cada noticia, se deben obtener todos los comentarios y la siguiente información de cada comentario: texto, autor, fecha y votos recibidos.
2. Requisitos para el almacenamiento y recuperación de información: Los datos obtenidos se deben poder almacenar permanentemente en disco para poder recuperarse posteriormente.
3. Requisitos para el análisis emocional de los textos: Utilizando los diccionarios de emociones se han de poder extraer y almacenar las emociones que transmiten las noticias.
4. Requisitos para la representación del análisis emocional: Con la información del análisis se deben poder crear diagramas que representen el impacto emocional de cada noticia.

3.2.2. Requisitos no funcionales

Además de los requisitos sobre la funcionalidad del sistema se necesitan cumplir con los siguientes requisitos para que el proyecto alcance un buen grado de calidad.

1. Requisitos de mantenibilidad: El código debe ser escrito de una manera modular, legible y bien comentado que faciliten en un futuro la corrección de posibles fallos y la reutilización y ampliación del software. El diseño de las estructuras para el almacenamiento de la información también debe ser modular.
2. Requisitos de fiabilidad: En el caso de ocurrir una situación anómala se debe poder mostrar el motivo de la misma. Si se trata de una situación anómala leve se deberá controlar y seguir con la ejecución normal de la aplicación.
3. Requisitos de rendimiento: La aplicación debe realizar las tareas eficientemente evitando realizar acciones innecesarias.

3.3. Análisis sobre la extracción de información

La programación web hoy en día está muy automatizada. Por eso, es de esperar que el código de las páginas web esté estructurado de una determinada manera. Sabiendo esto, será posible extraer toda la información de las noticias de manera automática. Para ello se necesitará una herramienta que sea capaz de descargar y manejar el código HTML de una página web.

3.3.1. Obtención de noticias a analizar

Se ha estudiado obtener la información de las noticias de tres maneras diferentes:

Manual: Introduciendo en el código de la aplicación las URL de las noticias que se desee obtener el análisis.

RSS: Gracias a la tecnología RSS se pueden obtener las últimas noticias publicadas por el periódico, por lo que se pueden extraer distintas noticias en función la hora de ejecución de la aplicación. Los enlaces a las noticias se obtienen del código de la página de RSS, entre las etiquetas “<guid>”. Estos son enlaces intermedios hacia la noticia y deberán de ser convertidos al enlace final.

Noticias “Lo más...”: Se trata de una sección específica del periódico 20minutos.es en el que se recopilan dentro de cuatro categorías las cinco noticias más importantes de cada una de estas. Es posible que una noticia que esté en una categoría se repita en otra, por lo que será necesario borrar las noticias duplicadas. Los enlaces a las noticias se encuentran en el código de la página principal. Cada categoría tiene sus enlaces dentro de una etiqueta de lista ordenada “ol” que tiene la clase “listado-noticias”.

Tras el análisis de los distintos modos de obtención de información del periódico, se determina que se utilizarán todos ellos en este trabajo.

3.3.2. Obtención de información de la noticia

En el caso de las noticias del diario online 20minutos, se observa que no hay mucha diferencia en el código HTML para dos noticias distintas. La siguiente imagen muestra en el apartado de la derecha las partes iguales entre el código de dos noticias mediante el color azul. El resto de colores indican las diferencias.

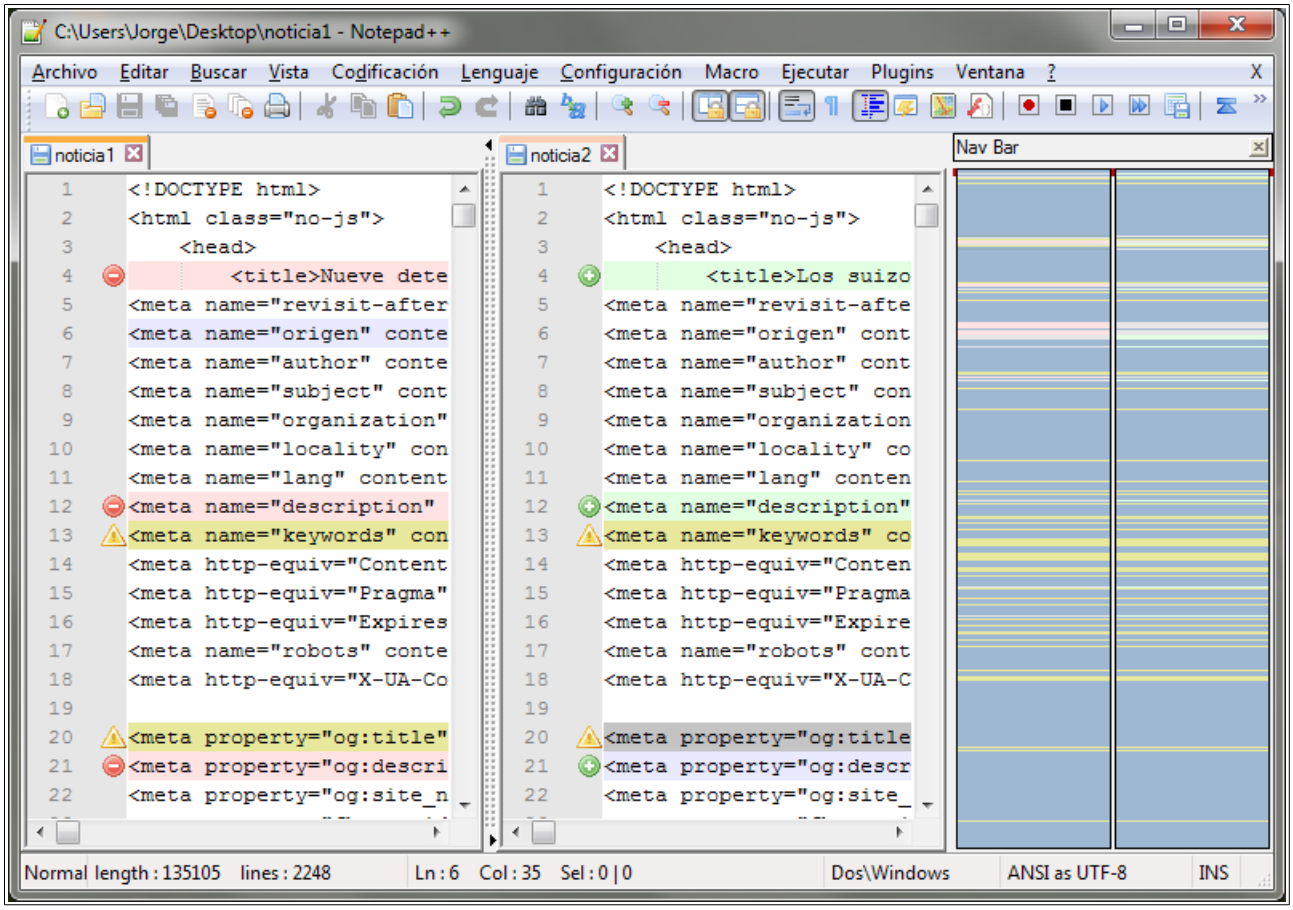

Figura 1. Diferencias entre dos noticias.

La información que se necesita extraer de cada noticia está siempre bajo las mismas etiquetas HTML, cambiando solamente el contenido entre unas noticias y otras.

Desde el código de una noticia se puede obtener el titular dentro de la etiqueta “title”.

El cuerpo de la noticia está contenido en la etiqueta “div” que tiene la clase “article-content”.

El autor y la fecha están almacenados en la etiqueta de lista no ordenada “ul” que tiene la clase “article-author”. Posteriormente se necesitarán separar para poderse almacenar de manera independiente.

Tras el análisis de cómo se almacenan las noticias y la información relacionada con cada una de ellas en el periódico, se concluye que es factible obtener la información deseada.

3.3.3. Obtención de información de los comentarios

Los comentarios se presentan en la parte inferior de la noticia mostrando los diez primeros. Para obtener todos los comentarios se necesitará navegar por las páginas de comentarios hasta haber extraído todos.

En el código se encuentran bajo una etiqueta cuyo id es “comentarios” y cada comentario bajo las etiquetas que tienen la clase “article-comment”. Dentro de estos elementos se pueden obtener el usuario que ha publicado el contenido mediante la clase “user”, el contenido del comentario con la clase “text textoCom” y el número de votos y la puntuación del comentario con la clase “button hits totalVotos”. Para obtener el número de votos positivos y negativos se han

de calcular a partir del número de votos y la puntuación del comentario.

Tras el análisis de cómo se almacenan los comentarios y la información asociada en el periódico, se concluye que es factible obtener toda esta información.

3.4. Análisis sobre el almacenamiento y recuperación de información

Se necesitará construir una estructura de datos que sea capaz de almacenar de manera temporal la información pertinente a las noticias y comentarios. Esta información será para las noticias el titular, la URL, el cuerpo de la noticia, el autor, los votos positivos y negativos recibidos, la fecha de publicación, el periódico de origen, la lista de comentarios para esa noticia y los datos relativos al análisis emocional. Los comentarios necesitarán guardar el texto, el autor, la fecha, los votos positivos y negativos recibidos, la noticia de origen y los datos del análisis emocional del comentario.

Una vez se haya obtenido y almacenado de manera temporal la información pertinente a las noticias y comentarios será necesario guardar esta información de forma permanente. Se necesitará una herramienta que sea capaz de llevar a cabo esta tarea y que además permita posteriormente recuperar esa información.

Una vez analizada esta tarea al detalle e identificados los requisitos relacionados con la gestión de la información (ver requisitos funcionales y no funcionales) se considera viable el proyecto en este sentido.

3.5. Reflexiones sobre el análisis emocional de los textos

Partiendo del trabajo y de los diccionarios mencionados anteriormente, existen cuatro emociones capaces de ser detectadas: joy (alegría), anger (enfado), fear (miedo) y sadness (tristeza). Las palabras que expresan cada una de estas emociones están distribuidas en un conjunto de doce diccionarios. Para cada emoción existen tres diccionarios en los que están clasificadas tres clases de palabras: “tal cual”, “sufijos” y “verbos”.

Las palabras que contienen los diccionarios expresan inequívocamente la emoción a la que está asignada. Para poder detectar que una palabra de un texto expresa una emoción será necesario comprobar si se encuentra en alguno de los diccionarios. Según el tipo de palabra será necesario añadir unas modificaciones u otras a las palabras de los diccionarios. Esto hará posible encontrar todas las posibles variaciones de la palabra del diccionario.

Se identifica, por tanto, la necesidad de determinar las modificaciones que se deben añadir a las palabras de cada diccionario para detectar las variaciones de las palabras contenidas en el mismo.

3.6. Reflexiones sobre la representación del análisis emocional

Una vez se hayan analizado los textos y obtenido los resultados del análisis emocional será necesario guardar esa información de una forma que facilite el estudio posterior de los resultados.

Se deberán crear gráficos y diagramas que muestren los datos del análisis. Para ello se utilizará una herramienta capaz de representar conjuntos de valores mediante gráficos de secciones y diagramas de líneas.

3.7. Tecnologías a utilizar

3.7.1. Lenguaje de programación

Hoy en día existen múltiples lenguajes de programación y para la mayoría de tareas casi todos son perfectamente válidos. No obstante, cada lenguaje tiene sus ventajas e inconvenientes a la hora de realizar un proyecto determinado. Para la realización de este proyecto se han planteado dos de los lenguajes más extendidos por su uso: los lenguajes C y Java. [7]

El lenguaje de programación C es el más extendido. Es un lenguaje de programación estructurado y su uso está más orientado hacia la eficiencia, pudiendo utilizar sentencias de más bajo nivel, optimizando el rendimiento del procesador. También brinda mucha libertad a la hora de realizar estructuras de datos aunque su manejo puede ser complejo. C es un lenguaje compilado, por lo que una vez se compile en un archivo ejecutable sólo funcionará para una determinada plataforma. No obstante, si se utilizara el estándar ANSI C se podría portar el código para ser compilado en otros sistemas.

Java es un lenguaje orientado a objetos. La principal ventaja es una mayor facilidad para definir y manejar estructuras de datos. La gestión de la memoria utilizada es automática, con lo cual se evitan posibles errores y se resta complejidad al desarrollar código. Al contrario que C, Java es un lenguaje interpretado por lo que sólo se necesita el intérprete en la máquina de destino para ejecutar el código desarrollado.

Para poder almacenar la información de periódicos, noticias y comentarios resulta más natural pensar en cada uno de ellos como objetos y la información de cada uno como cada uno de sus atributos. De esta manera parece muy razonable crear una estructura de datos para cada tipo de información. Java es un lenguaje orientado a objetos. En el caso de C, se podría utilizar C++, pero por los motivos expuestos anteriormente Java resultaría más adecuado para este proyecto, además de ser un lenguaje orientado al desarrollo de aplicaciones Web, por lo que facilitará después la interacción con la misma e incluso una potencial integración del desarrollo realizado en el periódico.

Por estas razones se ha elegido Java como lenguaje de programación.

3.7.2. Entorno de programación

Para desarrollar la aplicación se ha decidido utilizar un entorno de programación. Éstos son de gran utilidad y proporcionan muchas herramientas y facilidades a la hora de desarrollar código.

Dos de los entornos más utilizados son Netbeans y Eclipse. Los dos entornos son muy completos. Ambos permiten realizar proyectos de software en múltiples lenguajes de programación, añadir herramientas adicionales, creación automática de código y depuración de código.

Para la realización de este proyecto no hay una diferencia significativa entre las características de ambos entornos, sin embargo se ha escogido Netbeans por tener una interfaz más clara, útil y organizada que hará la programación un poco más intuitiva y eficiente.

3.7.3. Analizador de código HTML

Una de las fases del sistema a desarrollar consiste en obtener la información de las noticias. Para ello se necesita un analizador de código HTML que sea capaz de descargar y extraer la información que se encuentra en la noticia.

Se ha barajado utilizar uno de los siguientes analizadores: Jtidy, HTMLParser y Jsoup. Todos son librerías adicionales de Java, por lo que es posible añadirlos al proyecto y utilizar sus funciones de manera sencilla.

JTidy es un analizador de código HTML que permite comprobar si el código está bien formado. Además ofrece métodos de análisis del Modelo de Objetos del Documento o DOM. Esto permite navegar entre los elementos del código y extraer su información. Sin embargo Jtidy no tiene esta funcionalidad desarrollada en su totalidad.

HTMLParser permite obtener el código HTML a partir de una dirección URL. También ofrece métodos de análisis DOM de manera completa y permite acceder a toda la información del HTML.

JSoup incluye todas las funcionalidades de HTMLParser, pero además ofrece un nuevo método de búsqueda sobre los elementos del código HTML que simplifica la extracción. Con este método de búsqueda no es necesario extraer todos los elementos de un tipo y luego seleccionar el correcto sino que permite encontrar de manera directa el elemento buscado. Por este motivo se ha elegido utilizar JSoup frente a las otras herramientas para descargar, analizar y extraer la información de las noticias.

3.7.4. Almacenamiento de datos

Para tener una estructura y almacenar de manera temporal los datos se ha escogido el lenguaje Java que, como se ha detallado anteriormente, gracias a que es un lenguaje orientado a objetos permite estructurar los datos de manera sencilla.

Para almacenar esta información de manera permanente se ha pensado inicialmente en bases de datos relacionales. La estructura de estas bases de datos es muy similar a los objetos en Java, pudiendo tener tablas que representen a las clases, que cada entrada de la tabla sea un objeto y que sus relaciones entre objetos sean nuevas columnas de las tablas. Además, las bases de datos permiten realizar búsquedas muy detalladas mediante la realización de consultas.

No obstante, manejar los datos de una base de datos relacional es más complejo que manejar los datos de objetos en Java. A su vez, la dificultad de la correcta instalación y configuración de la base de datos así como de la integración en la aplicación no es comparable al uso inmediato de los objetos en java.

Por ello, se ha pensado en otra solución a la persistencia de los datos y es en vez de guardar los datos existentes en una base de datos guardarlos en un fichero, de manera que sea posible recuperarlo en cualquier momento. Una herramienta que permite realizar esto es XStream. Se trata de una librería para Java que se encarga de transformar los objetos que se le indiquen a un fichero con estructura XML. De esta forma es posible leer los datos que se han guardado mediante un editor de texto o un navegador web. Para obtener nuevamente la información no es necesario

cambiar la manera de operar con los datos. XStream puede convertir el contenido del fichero en los mismos objetos guardados previamente. Aparte de ser una herramienta más sencilla que las bases de datos, tiene otra gran ventaja y es que se pueden extraer y almacenar los ficheros que se vayan generando y utilizarlos según convenga en la parte de análisis de noticias, sin tener que interactuar con ningún gestor de bases de datos.

Las bases de datos son unas herramientas muy útiles y con muchas funcionalidades, pero en este caso sólo se necesita una pequeña cantidad de ellas. Por lo tanto se ha optado por la herramienta XStream, que aporta todas las funcionalidades necesarias de manera mucho más sencilla.

3.7.5. Representación de datos

Las herramientas de representación de datos analizadas han sido JavaFX Charts, XChart, y JFreechart.

JavaFX Charts es una herramienta que permite crear y mostrar varios tipos de gráficos y diagramas. Es un paquete incluido en el JDK de Java, por lo que no es necesario incorporar ninguna librería adicional. Sin embargo no se pueden exportar los diagramas creados de manera directa.

Otra herramienta de creación de diagramas es XChart. Se trata de una librería para Java muy ligera que ofrece de varios tipos de diagramas. Es fácil de usar y permite exportar los diagramas a una imagen de manera sencilla. No obstante, el número de tipos de diagramas que se pueden realizar es algo limitado.

JFreeChart es la herramienta elegida para representar los datos del análisis emocional. Se trata de una librería muy potente y configurable a la vez que fácil de utilizar. JFreeChart ofrece una amplia variedad de diagramas en los que se pueden variar la forma y los colores de representación de los datos. Además existe la posibilidad de exportar a ficheros de imagen los diagramas creados.

4.Diseño

En esta sección se describirá la arquitectura general del sistema. A continuación se mostrará la estructura de datos diseñada detalladamente. Finalmente se describirá el diseño detallado de los módulos.

4.1.Arquitectura general del sistema

El sistema está compuesto de dos módulos principales. El primer módulo lee la información relativa a cada noticia y la procesa. El resultado es un fichero en XML con toda la información de las noticias leídas y los comentarios asociados. El segundo módulo lee ese fichero y los diccionarios de emociones y con ellos se realiza el análisis emocional. Tras este análisis genera unos diagramas con la representación del análisis emocional de cada noticia.

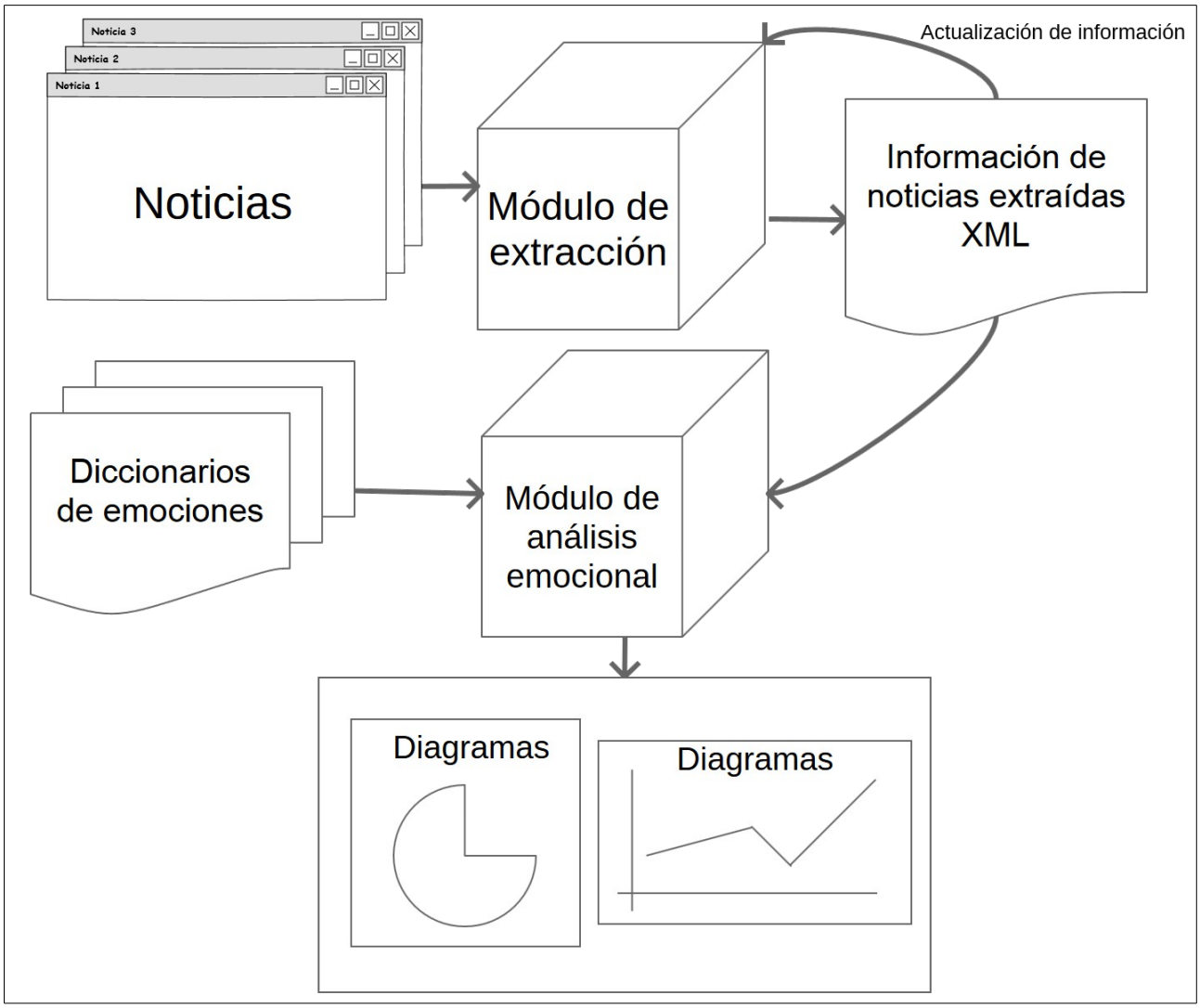

Figura 2. Diagrama general del sistema.

4.2. Estructura de datos general del sistema

Para realizar el diseño de la estructura de datos del sistema se aprovecharán las ventajas que ofrece el lenguaje de programación elegido. Al ser Java un lenguaje orientado a objetos la estructura de datos del sistema se ha organizado en varias clases. Cada módulo del sistema cuenta con una lista de objetos que almacenan toda la información para cada periódico. A su vez, estos objetos tienen una lista de noticias relativas a ese periódico con toda su información asociada. Cada noticia tiene además una lista con todos los comentarios de esa noticia y su información pertinente. Además tanto las noticias como los comentarios tienen un objeto en el que se almacena toda la información relativa al análisis emocional.

A continuación se muestra el diagrama de clases que cada módulo implementará y la descripción de cada clase.

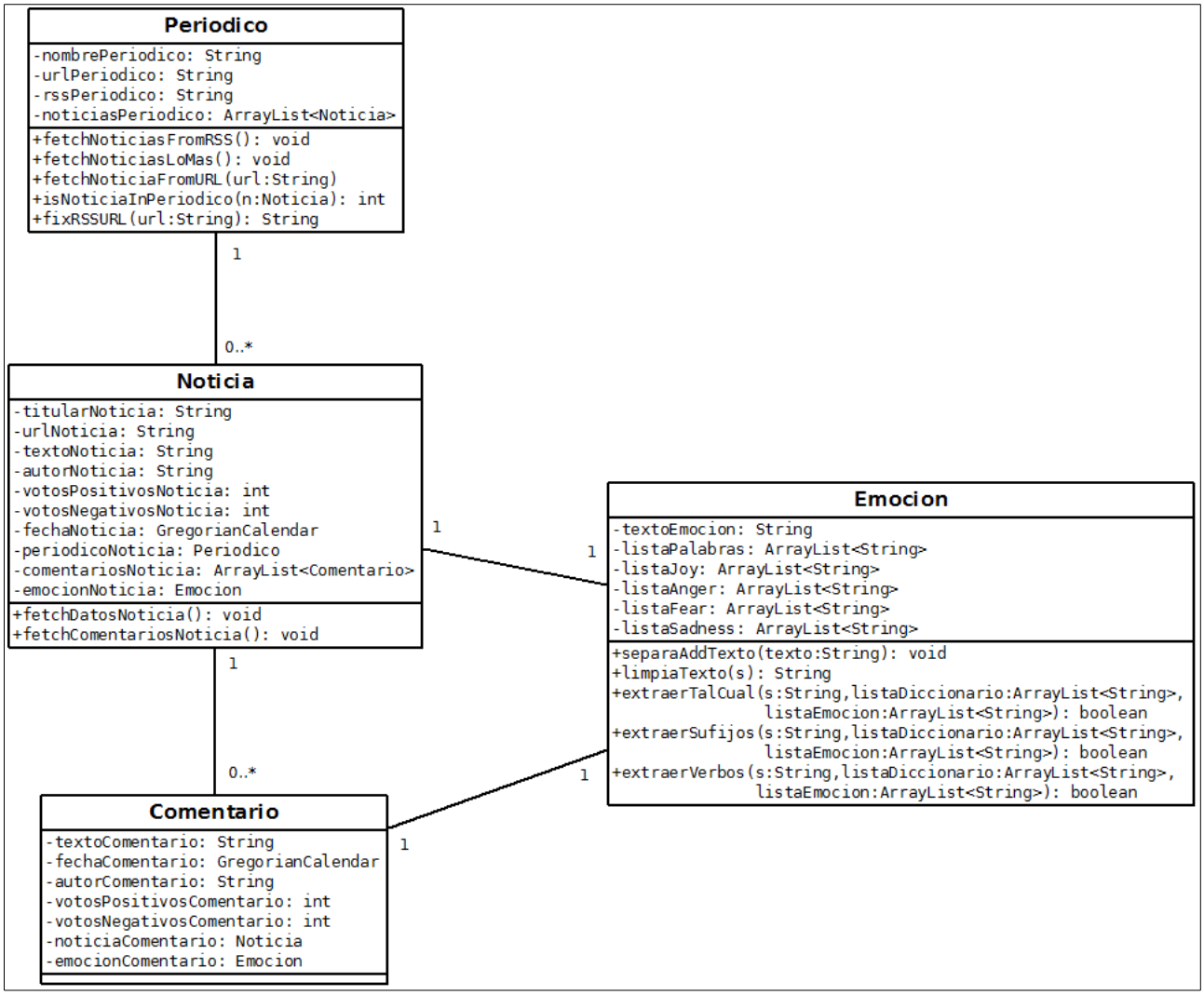

Figura 3. Diagrama de clases del sistema.

4.2.1. Clase Periódico

nombrePeriodico: El nombre identificativo del periódico.

urlPeriodico: La dirección web principal o URL del periódico.

rssPeriodico: La dirección donde están disponibles las últimas noticias de ése periódico.

noticiasPeriodico: Una lista de objetos de tipo Noticia con todas las noticias de ese periódico.

4.2.2. Clase Noticia

titularNoticia: El titular de la noticia.

UrlNoticia: La dirección web o URL de la noticia.

TextoNoticia: El cuerpo de la noticia.

AutorNoticia: El autor de la noticia.

VotosPositivosNoticia: Los votos positivos recibidos para esa noticia.

VotosNegativosNoticia: Los votos negativos recibidos para esa noticia.

FechaNoticia: La fecha de publicación de esa noticia.

PeriodicoNoticia: El objeto Periódico al que pertenece esa noticia.

ComentariosNoticia: Una lista de objetos de tipo Comentario con todos los comentarios de la noticia.

EmocionNoticia: El objeto que almacena la información relativa a las emociones de esa noticia.

4.2.3. Clase Comentario

TextoComentario: El contenido del comentario.

FechaComentario: La fecha de publicación del comentario.

autorComentario: El autor del comentario.

votosPositivosComentario: Los votos positivos recibidos para ese comentario.

votosNegativosComentario: Los votos negativos recibidos para ese comentario.

noticiaComentario: El objeto Noticia al que pertenece ese comentario.

emocionComentario: El objeto que almacena la información relativa a las emociones de ese comentario.

4.2.4. Clase Emoción

TextoEmocion: El texto del comentario o de la noticia a ser analizado.

ListaPalabras: Una lista con cada palabra del texto de la emoción

listaJoy, listaAnger, listaFear y listaSadness: Lista de palabras encontradas para cada emoción tras realizar el análisis emocional.

4.3. Módulo de extracción

En este módulo se dispondrá de una lista de objetos de periódicos en los que se almacenará y manejará toda la información. Después se realizarán una serie de tareas consistentes en la

obtención de las noticias y los comentarios y el almacenamiento en disco de los datos. El siguiente diagrama (figura 4) representa los pasos principales de este procedimiento. En los siguientes apartados se explican los detalles de cada uno de estos pasos.

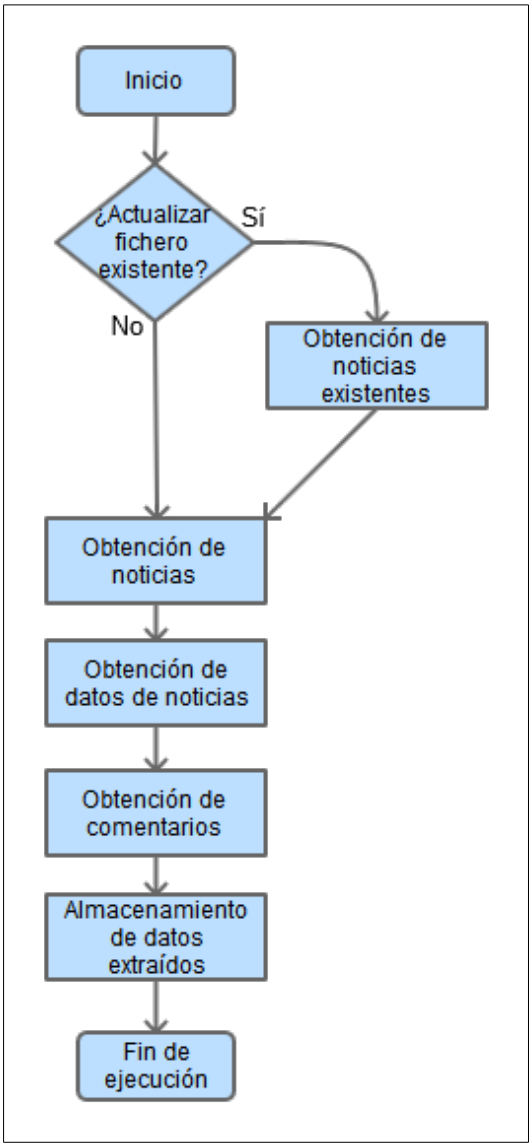

Figura 4. Diagrama del módulo de extracción.

4.3.1. Actualización de noticias existentes

Durante la actualización de noticias, si ya existe un fichero con noticias extraídas anteriormente se puede leer de él, obtener las noticias, actualizarlas y añadir otras noticias después en vez de crear un fichero nuevo.

Una variable indicará si se desea actualizar un fichero existente o por el contrario se prefiere crear uno nuevo.

En el caso que se desee actualizar se deberán leer del fichero los datos de los periódicos y las noticias. Si su lectura resulta correcta se tendrá en la lista de objetos de periódicos los datos leídos, listos para añadir nuevos datos o ser actualizados. En caso contrario se añadirá a la lista un nuevo periódico con la información de 20minutos.es para proceder a leer las noticias y sus datos.

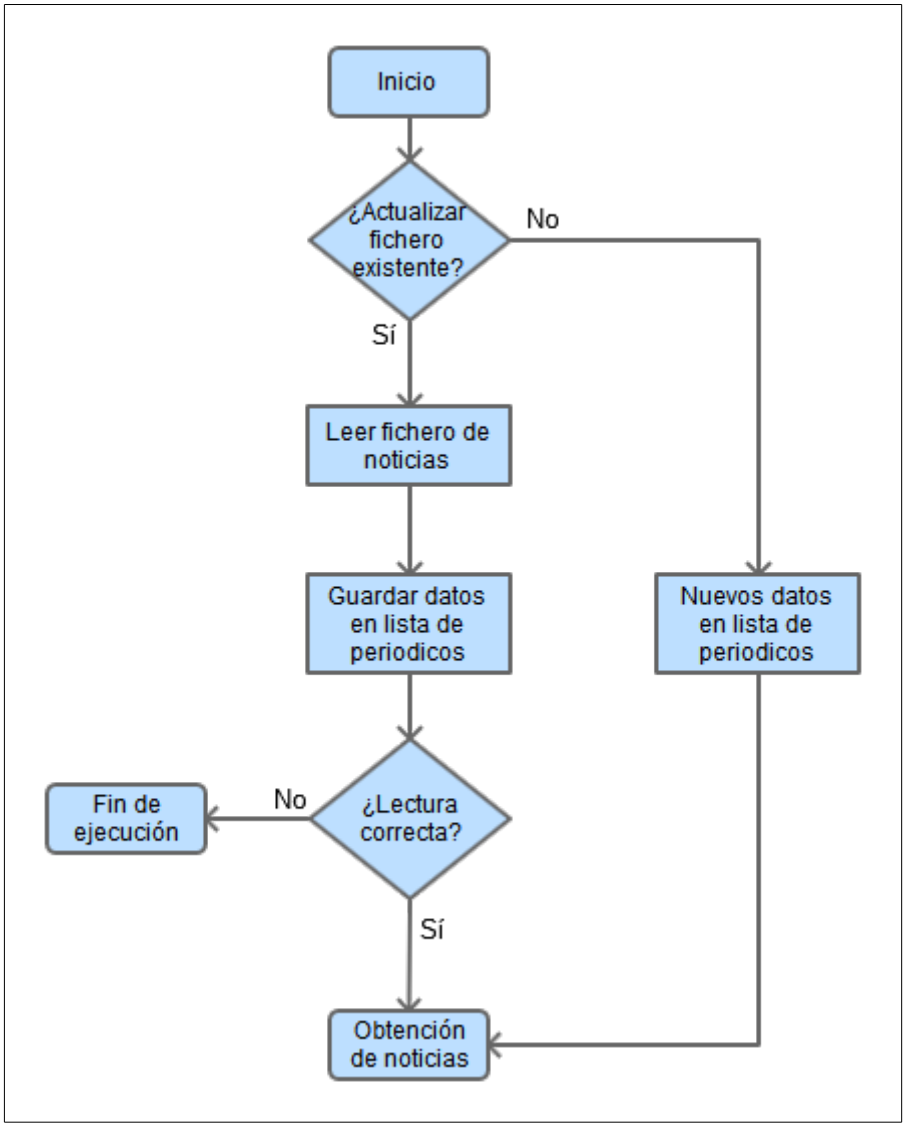

Figura 5. Diagrama de actualización de noticias.

4.3.2. Obtención de noticias

Durante la fase de obtención de noticias, para cada periódico, se obtienen las noticias. La clase Periódico tendrá tres métodos distintos para realizar esa tarea.

- fetchNoticiasFromURL(URL):

Este método comprobará que la URL es válida realizando una conexión con la librería Jsoup. También comprobará si la noticia no estaba incluida anteriormente para evitar noticias duplicadas. Para ello mirará si alguna noticia ya almacenada tiene la misma URL. Si resulta que ya existe una noticia con esa URL no hará nada y continuará de nuevo con la ejecución. Si la noticia es válida y no está repetida creará una con esa URL y se añadirá a la lista de noticias del periódico. La figura 6 muestra el diagrama correspondiente a este proceso.

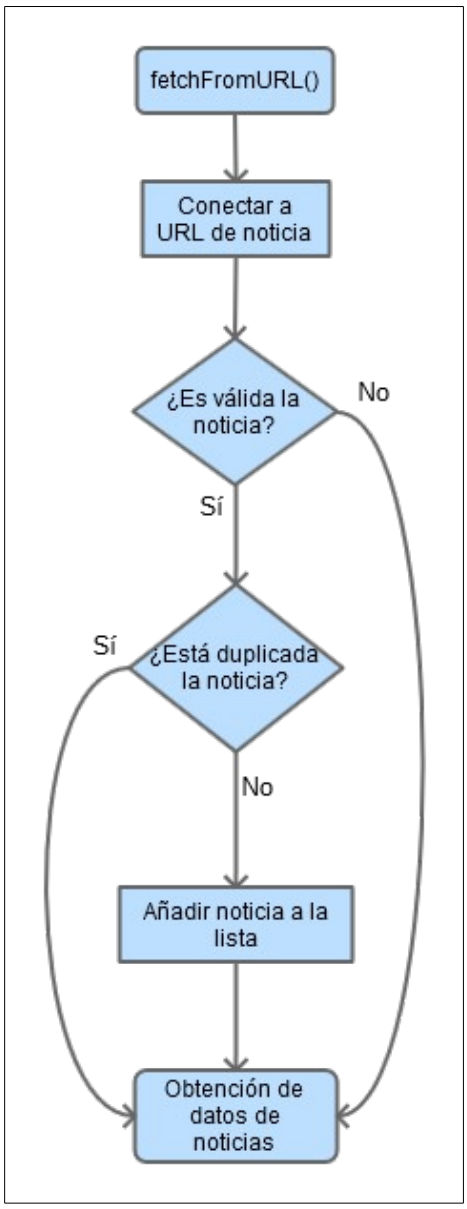

Figura 6. Diagrama flujo del método fetchFromURL().

- fetchNoticiasLoMas():

En la página principal del periódico 20Minutos.es existe un apartado con las noticias más destacadas según cuatro criterios denominados “Lo más”: ECO, Visto, Valorado y Comentado. Este método obtendrá esas noticias y las incluirá en la lista de noticias del periódico. La figura 7 muestra el diagrama correspondiente a este proceso.

Lo primero que realizará es una conexión con Jsoup a la URL de la página del periódico para obtener el código HTML. Navegando entre las etiquetas del código con Jsoup se pueden encontrar los enlaces a las noticias. Las URL que se necesitan se encuentran bajo las etiquetas “ol” de clase “listado-noticias”. Los enlaces se encuentran en el valor del atributo “href” de los elementos de etiqueta “a”. Por último hay que filtrar los resultados que no contengan en la URL “#social” ya que la URL de esas noticias no son compatibles.

Una vez se tenga la URL de esa noticia se comprueba que no exista en ese periódico y se añade a la lista de noticias del periódico. En el caso de que exista se procederá con la siguiente URL.

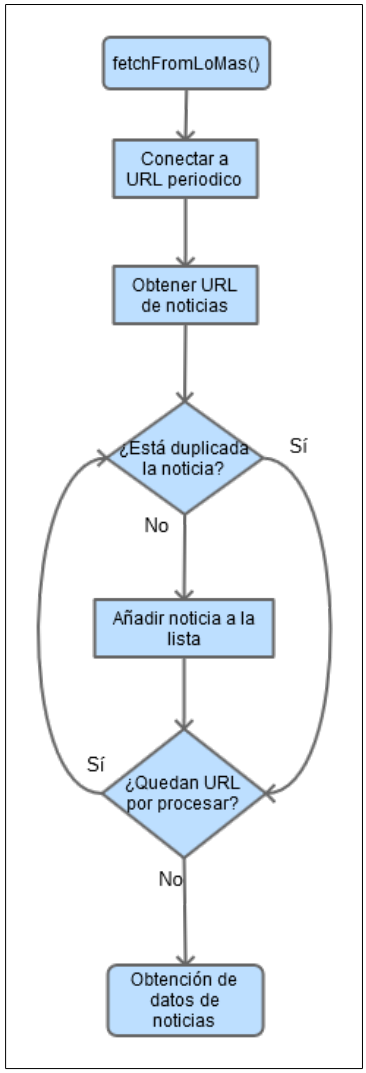

Figura 7. Diagrama de flujo del método fetchFromLoMas().

- FetchNoticiasFromRSS():

Una forma de automatizar el proceso de extracción de nuevas noticias es obtener las noticias de los RSS del periódico. A través de esta página RSS se podrán obtener las últimas noticias cada vez que se acceda a ella.

Para ello primero se realiza una conexión con Jsoup a la URL de RSS del periódico para obtener el código HTML de los RSS. Las URL de las noticias están bajo las etiquetas "guid". Sin embargo, estas URL son distintas a las URL de una noticia estándar, ya que conducen a una página intermedia con publicidad. A continuación se muestra la URL intermedia y la URL final de una noticia:

URL intermedia: http://20minutos.feedsportal.com/c/32489/f/478284/s/3bbcf253/sc/5/l/0L0S20Aminutos0Bes0Cnoticia0C21735650C0A0Cinvestigacion0Ematerial0Csoporta0Esu0Epeso0Cmicrofabricacion0C/story01.htm

URL final: http://www.20minutos.es/noticia/2173565/0/investigacion-material/soporta-su-peso/microfabricacion/

A partir de esta URL se puede obtener la URL de la noticia original. Es necesario eliminar la primera parte del enlace hasta la primera palabra “minutos”. Después se debe añadir “http://www.20” por delante de “minutos”. Finalmente se debe sustituir las siguientes apariciones de cadenas “0A”, “0B”, “0C”, y “0E” por “0”, “.”, “/”, y “0E” respectivamente. Este arreglo lo realizará un método llamado fixRSSURL al que se le pasará la URL intermedia como parámetro y devolverá la URL final.

Por último se comprobará que la noticia no exista previamente y se añadirá a la lista de noticias del periódico. La figura 8 muestra el diagrama correspondiente a este proceso.

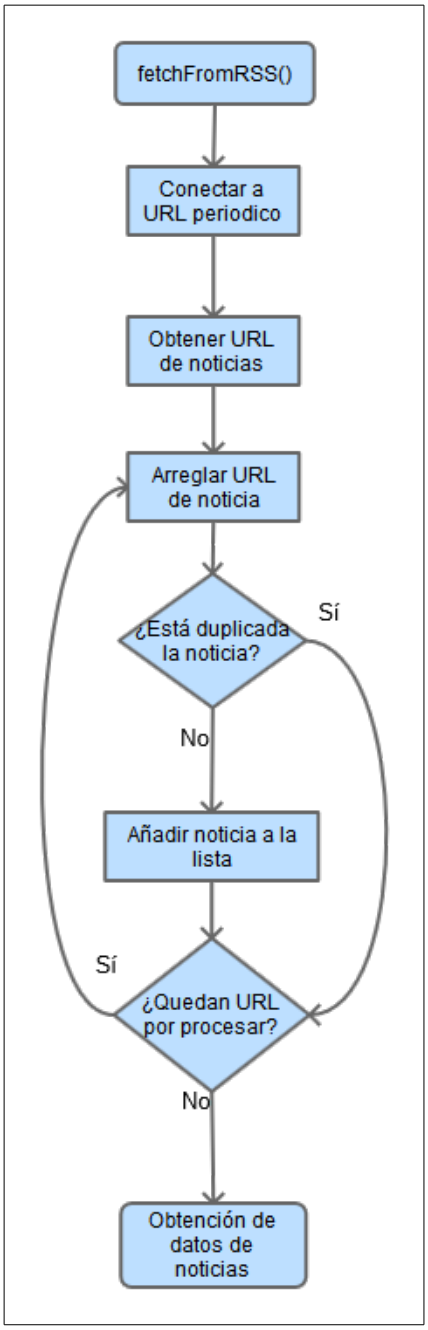

Figura 8. Diagrama de flujo del método fetchFromRSS().

4.3.3. Obtención de datos de las noticias

Tras añadir las noticias mediante uno o varios de los métodos anteriores se procederá a obtener los datos de cada noticia. El método fetchDatosNoticia se encargará de realizar esa tarea. La figura 9 muestra el diagrama correspondiente.

Primero se obtendrá el código HTML de la noticia para poder ser analizado. En él se encuentra toda la información necesaria. Se deberá extraer de él el titular, el texto el autor, la fecha, la puntuación y los votos. El título se encuentra bajo la etiqueta "title", el cuerpo de la noticia bajo la clase "article-content" y el autor y la fecha bajo la clase "article-author". La puntuación y los votos están bajo el elemento con la etiqueta "span" de clase que empieza por "button hits".

El autor y la fecha aparecen juntos, por lo que se deben separar. El formato de esta cadena de caracteres sigue este patrón: "Autor Fecha (Hora) (@usuarios Twitter)", donde la hora y los usuarios de twitter pueden aparecer o no.

Autor y fecha	EFE. 17.05.2014
Autor, fecha y hora	EDU CASADO. 17.05.2014 - 18:53h
Autor, fecha y usuario de twitter	EDU CASADO 17.05.2014 @educasado
Autor, fecha, hora y usuario de twitter	CLARA HDEZ. 10.05.2014 - 21:44h @clarittis Google+

Tabla 1. Ejemplos de autor y fecha del código HTML

Se sabe que el autor está al inicio de la cadena, pero no se sabe su longitud para poder extraerlo. La fecha, con hora o sin ella, tiene una longitud definida, pero su posición en la cadena depende de la longitud del nombre del autor, dato que se desconoce. Tampoco se conoce la longitud de los usuarios de twitter, pero se sabe que siempre empiezan con una "@" y este carácter siempre estará al final de la fecha o de la hora. Si se incluye la hora, el carácter que se encuentra a dos posiciones a la izquierda de la "@" será una "h". Si no es una "h" significa que esa fecha no incluye la hora. Como la fecha tiene una longitud fija, se puede extraer a partir del carácter de la @.

En el caso que no existan usuarios de twitter, se puede comprobar si la fecha incluye hora, comprobando que el último carácter de la cadena sea una "h". En caso contrario no se incluye la hora.

Para obtener los votos positivos y negativos recibidos se han de calcular a partir de los votos recibidos y de la puntuación, ya que la página sólo muestra estos datos. Los votos positivos y negativos se pueden obtener tras resolver el siguiente sistema de ecuaciones, del que se conocen los datos de la puntuación y del número de votos:

$$Puntuación = Votos\ positivos - Votos\ negativos$$

$$Número\ de\ votos = Votos\ positivos + Votos\ negativos$$

Ejemplo: Noticia con puntuación +4 y con 8 votos recibidos. Si despejamos los votos positivos en la primera ecuación obtenemos "Votos positivos = 4 + Votos negativos". Ahora despejamos en la segunda ecuación los votos negativos y tenemos "Votos negativos = 8 – Votos positivos". Ahora si en esta ecuación insertamos el valor obtenido de Votos positivos se obtiene que los votos

negativos son 2. Por lo tanto usando este valor en la primera ecuación obtenida tenemos que los votos positivos son 6.

A continuación se preparará el texto para el análisis emocional. Para ello primero se limpiará el texto de todo tipo de tildes, dejando sólo las vocales. Después se eliminarán todos los caracteres que no representen letras, incluyendo entre éstos los dígitos, y se dejarán todas las letras en minúscula. Por último se separarán todas las palabras y se guardarán en una lista de palabras en la Clase Emoción asociada a la Noticia.

Todos estos datos extraídos serán almacenados en sus respectivos elementos asociados de la noticia.

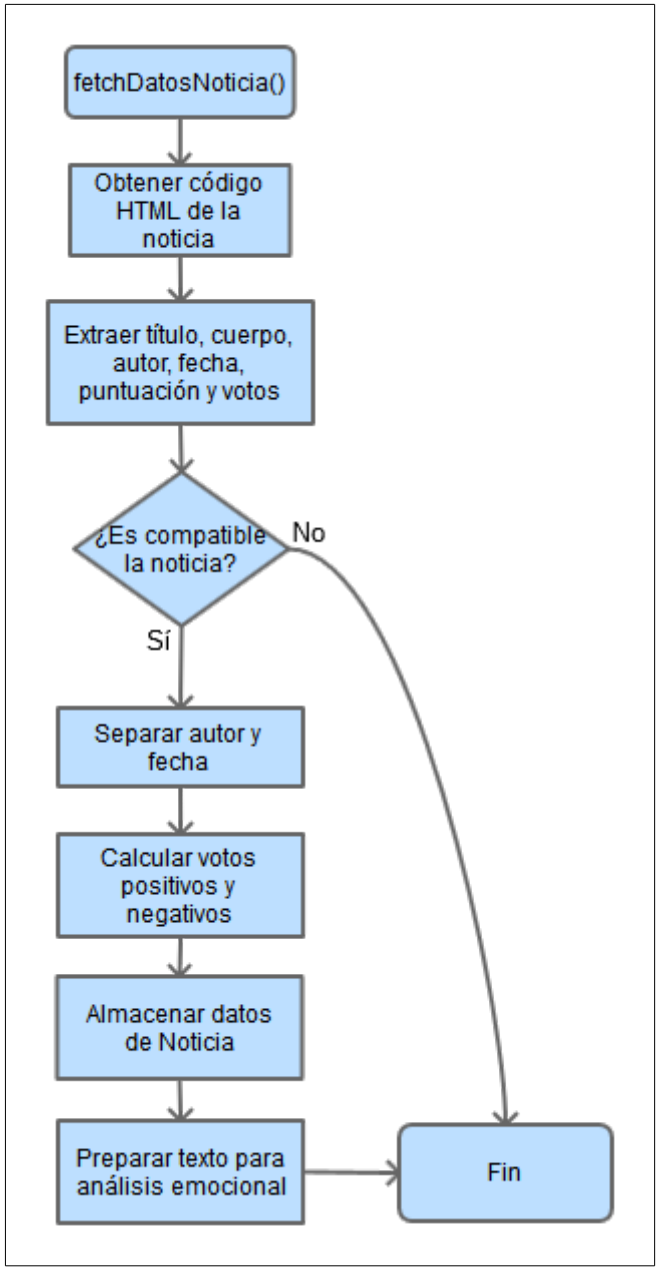

Figura 9. Diagrama de obtención de noticias.

4.3.4. Obtención de los comentarios de las noticias

Una vez que cada noticia tenga su información almacenada el paso siguiente será obtener los comentarios de esa noticia. La figura 10 muestra el diagrama correspondiente a este proceso.

Primero se obtendrá el código HTML de la noticia. En función del número de comentarios será necesario ir a las siguientes páginas ya que cada página sólo muestra diez comentarios. El número de comentarios se encuentra en el elemento que tiene por id “commentcount”. A partir de este número se puede calcular el número de páginas que se necesitan analizar. En principio, si la noticia no tiene comentarios, o bien, si se está actualizando una noticia y se comprueba que el número de comentarios obtenidos previamente es mayor a los de la noticia a analizar se pasará a analizar la siguiente noticia. A partir de este punto se sabe que se han de obtener nuevos comentarios. Para diferenciar entre una noticia antigua o nueva se comprobará si la noticia tiene algún comentario almacenado. En caso afirmativo una variable almacenará que la noticia es nueva o por el contrario que es antigua.

El número de páginas se calcula realizando la división entera del número de comentarios entre diez, y se suma uno si el número de comentarios no es múltiplo de diez. Es decir, de uno a diez comentarios sería una página, de once a veinte serían dos y así sucesivamente.

El valor que indica el número de página en la URL es el número que aparece en la posición número 40 de la URL, siendo la primera página la cero.

Página 1	http://www.20minutos.es/noticia/2141665/**0**/sequia/intensa/este-sur-espana/
Página 5	http://www.20minutos.es/noticia/2141665/**4**/sequia/intensa/este-sur-espana/
Página 11	http://www.20minutos.es/noticia/2141665/**10**/sequia/intensa/este-sur-espana/

Tabla 2. Ejemplos de URL de páginas de comentarios

Para cada página se obtendrán todos los comentarios y su información. Al terminar se obtendrá la URL de la siguiente página para proceder con los demás comentarios. Esto se realizará reemplazando el número de página de la URL. Este número puede tener varias cifras, pero como se sabe el valor de la página actual se puede calcular fácilmente e inmediatamente sustituirlo por el siguiente.

En cada página todos los comentarios están bajo el elemento con id “comentarios”. Dentro de éste está cada comentario que tiene la clase “article-comment”. En cada uno de éstos elementos se encuentran el autor y la fecha del comentario en los elementos con clase “user” y la puntuación en los elementos cuya etiqueta sea “span” con clase “button hits totalVotos”. El autor del comentario y la fecha aparecen en el mismo elemento, por lo que tendrán que separarse. Además el autor aparece por duplicado. Como la fecha del comentario tiene un tamaño fijo y siempre está al final del elemento será de fácil extracción. Para extraer el autor del comentario se deberá eliminar la parte donde aparece la fecha y dividir su longitud entre 2.

Los comentarios, al igual que las noticias, pueden recibir votos positivos y negativos. El cálculo de esta información es idéntico a la detallada durante la extracción de datos de las noticias.

De igual modo que las noticias, para cada comentario se preparará el texto para el posterior análisis emocional.

Si la noticia es nueva se guardará toda la información y se añadirá el comentario a la lista de comentarios de la noticia. En caso de que sea una noticia antigua se comprobará la fecha del comentario actual con la del último comentario de la noticia y sólo en ese caso se añadirá el comentario a la lista.

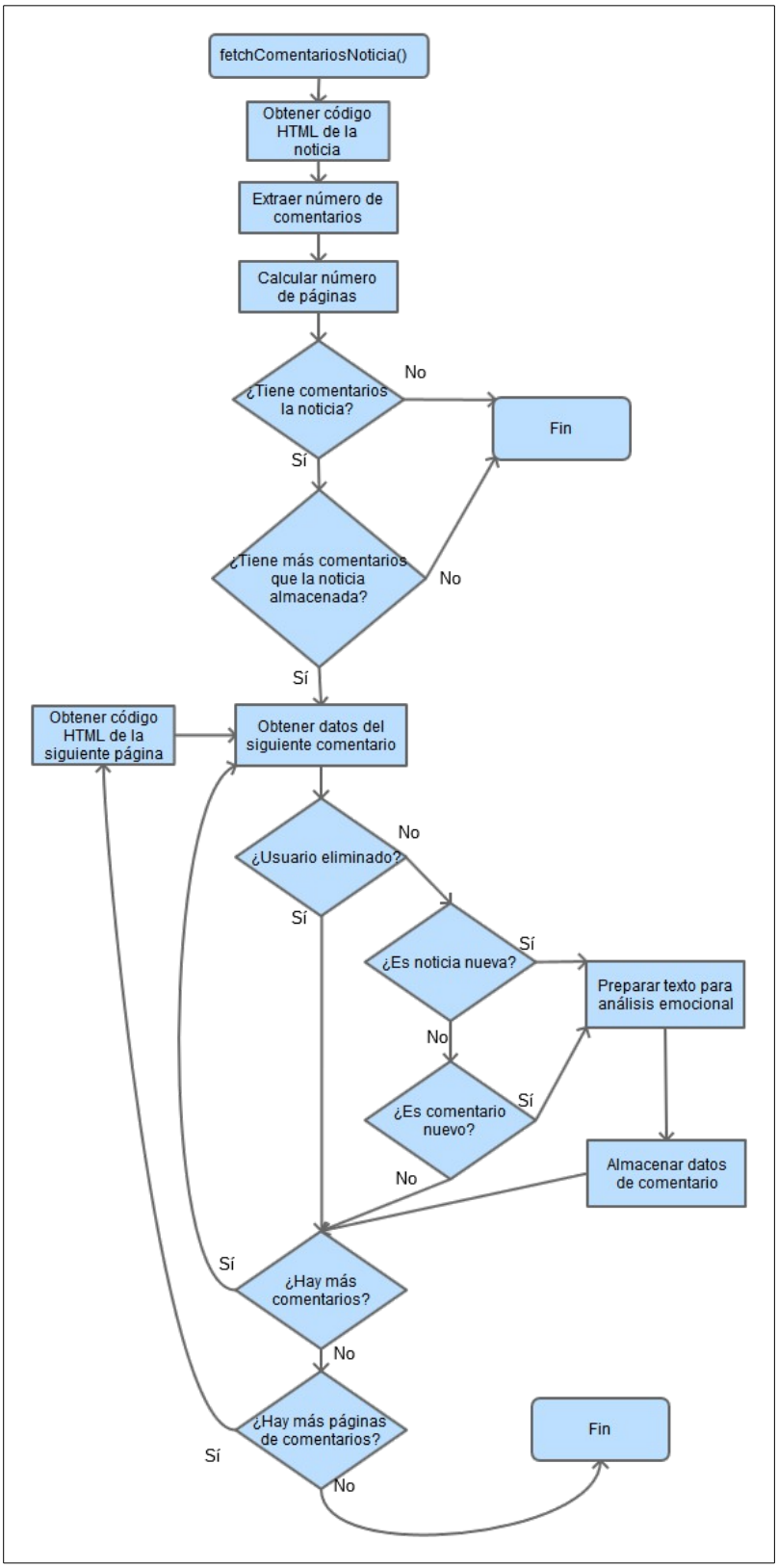

Figura 10. Diagrama de obtención de comentarios.

4.3.5. Almacenamiento de los datos extraídos

Llegados a este punto, se tiene en la memoria una lista de cada periódico con todas las noticias y de cada una de ellas todos los comentarios.

Debido a noticias que no siguen una estructura estándar, es posible que se hayan obtenido noticias que le falten algunos datos. Por ejemplo, una noticia de tipo reportaje fotográfico no tiene cuerpo de noticia, por lo que no podrá ser analizada. En este caso las noticias deberán ser eliminadas.

El último paso será guardar la lista de periódicos con toda la información en un fichero de manera permanente. La lista será procesada por una herramienta que la transformará a texto de manera consistente para su posterior lectura. El formato de salida será un fichero XML al que se le añadirá una cabecera para permitir su lectura de manera cómoda.

La librería XStream permite convertir los objetos con los datos de los periódicos a un texto con estructura XML. A este texto se le añade una cabecera XML que permite su fácil lectura. Finalmente se guarda en un fichero llamado “noticias.xml” para su posterior uso en el módulo de análisis emocional.

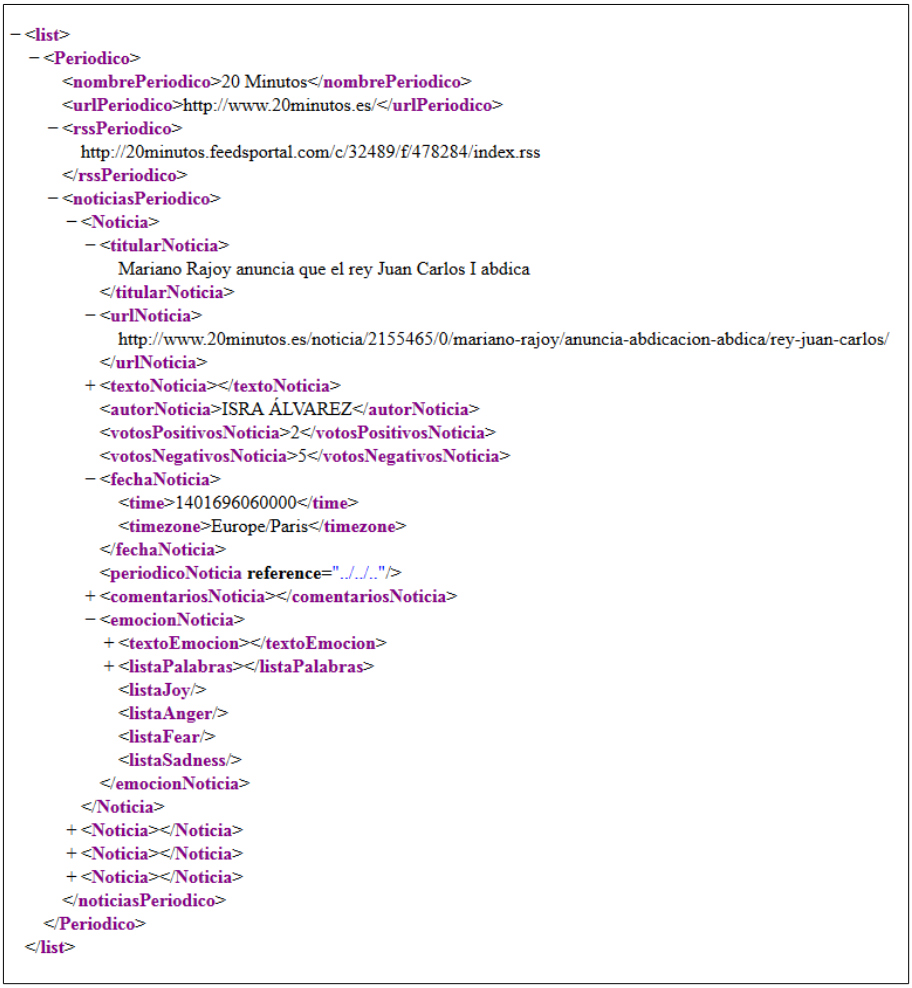

Figura 11. Fragmento del fichero noticias.xml.

4.4. Módulo de análisis emocional

En este módulo se recuperará la información extraída en el módulo anterior almacenándola en una lista de objetos de Periódicos. Se leerá la información de los diccionarios de emociones en listas de palabras y se procederá a realizar el análisis emocional de las noticias y de los comentarios. Por último se crearán diagramas que representan los datos analizados previamente. La figura 12 muestra la secuencia de pasos a realizar en esta etapa.

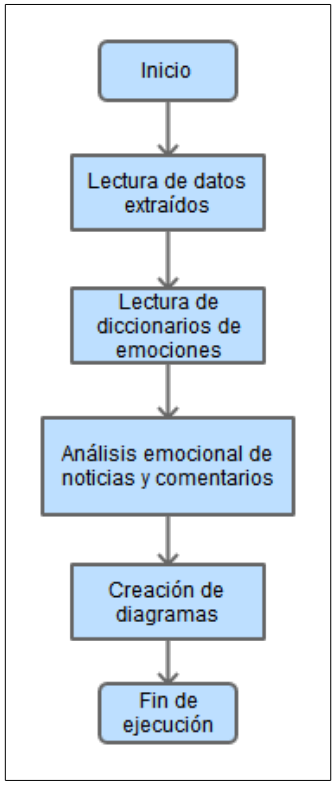

Figura 12. Diagrama del módulo de análisis.

4.4.1. Lectura de datos extraídos

Lo primero que realizará este módulo es una lectura del fichero guardado mediante el módulo de extracción utilizando la misma herramienta de la manera inversa. En el caso que el fichero no exista, o bien los datos no sean consistentes debido a una manipulación manual del fichero, o bien se trate de un fichero distinto se mostrará un mensaje informando del error y se cerrará.

4.4.2. Lectura de diccionarios de emociones

El siguiente paso en este módulo será leer los diccionarios de emociones. El método leeDiccionario() se encargará de ello. Este método leerá de un fichero cuyo nombre será pasado como parámetro y almacenará las palabras leídas de él en una lista de palabras también pasadas como parámetro. Este método se ejecutará tantas veces como diccionarios haya, cada uno con su nombre de fichero y su lista de palabras asociada.

Los diccionarios se tratan de ficheros de texto en los que cada línea contiene una palabra. Hay tres tipos de diccionarios según el tipo de palabras para cada emoción y cuatro emociones

distintas, por lo que hacen un total de doce diccionarios y doce listas de palabras.

4.4.3. Análisis emocional de las noticias y comentarios

Llegados a este punto se tienen por un lado las listas de las palabras limpias de los textos de las noticias y de los comentarios y por otro las palabras de los diccionarios guardados en otras listas de palabras. Con estas dos partes se realizará el análisis emocional de las noticias y comentarios.

El análisis de un texto se realizará de la siguiente manera: cada palabra del texto será comparada con las palabras de cada diccionario. Si en algún momento hay una coincidencia se almacenará la palabra del texto en una lista de palabras del mismo tipo de emoción que el diccionario en el que se ha detectado y se procederá a comparar la siguiente palabra del texto. En el caso en el que no se detecte en ningún diccionario no se hará nada y se pasará a comprobar la siguiente palabra del texto. Finalmente se tendrán cuatro listas con palabras del texto detectadas, una lista por cada emoción. El análisis se realizará tanto a los textos de las noticias como a todos y cada uno de los comentarios de cada noticia.

Las palabras en general pueden variar y seguir transmitiendo la misma emoción, como por ejemplo cuando varían en género o en número. Para evitar no detectar esos casos se necesitará comprobar la palabra del texto con todas las variaciones de la palabra del diccionario. Para evitar abarrotar los diccionarios de posibles variaciones de las palabras y mantenerlo simple los diccionarios contienen la base o raíz de la palabra. Sólo se necesitará añadir texto al final de esa palabra para formar otras nuevas. Cuando una palabra del texto sea comparada con una palabra de un diccionario se añadirán a la del diccionario todas las posibles variaciones.

No obstante, no todas las palabras tienen las mismas variaciones. Por ejemplo, las variaciones de un verbo con todas sus conjugaciones son distintas a las de un sustantivo. Por ese motivo las palabras de cada emoción están clasificadas en tres diccionarios. Debido a que no se puede saber si una palabra necesitará cambiar la raíz estos diccionarios contienen la raíz de la palabra con la que poder añadir terminaciones y formar nuevas palabras. Si una palabra tiene varias raíces aparecerá tantas veces como raíces tenga. Por ejemplo "audaz" y "audaces" aparecerá en los diccionarios como "audaz" y "audac". Aunque de esta manera se forman palabras que no existen no supone un problema puesto que no aparecerán en el texto a analizar y además se conseguirá detectar todas las variaciones válidas en el texto.

El primer diccionario, denominado "Tal Cual", contiene las palabras que no varían a excepción del número y necesiten agregar "es" para formar el plural.

El segundo es denominado “sufijos” y aparecen las palabras que pueden variar mediante la incorporación de sufijos. Cuando se compare la palabra del texto con las palabras de este diccionario se le incorporarán a estas últimas los sufijos de la tabla siguiente.

Género	o, a, os, as, e
Número	s, es
Aumentativos	azo, azos, aza, azas, on, ones, ona, onas, ote, otes
Disminutivos	ito, itos, ita, itas, cito, citos, cita, citas, ecito, ecitos, ecita, ecitas
Despectivos	astro, astra, ucho, ucha, uchos, uchas, acho, acha, achos, achas
Cualidad	az, aces, ería, erias, eza, ia, ias, idad, idades, tud, tudes, ura, uras, dad, dades, ez, eces, icia, icias, or, ores
Acción	anza, anzas, ato, atos, dura, duras, ia, ias, mento, mentos, aje, ajes, ata, atas, azgo, azgos, cion, ciones, do, dos, eria, erias, miento, mientos, toria, torias
Colectividad	amenta, amentas, eria, erias, ado, ados, al, ales, erio, erios, io, ios
Superlativos	isimo, isima, errimo, errima
Adjetivantes	oso, osa, osos, osas, able, ables, ible, ibles, enco, encos, enca, encas, ante, antes, ente, entes, ido, ida, idos , idas, ano, ana, anos, anas ado, ada, ados, adas, ento, enta, entos, entas, izo, iza, izos, izas, oide, oides
Adverbializantes	mente
Nominalizantes	ancia, ancias, encia, encias, anza, anzas, cion, ciones, sion, siones, ismo, ismos, dad, ,dades, tad, tades, ada, adas, eria, erias, aje, ajes, ez, eces, mento, mentos, miento, mientos, dura, duras
Verbalizantes	ar, ear, ificar, izar, ecer

Tabla 3: Sufijos [8]

El último diccionario es el de “verbos”. En éste aparecen las formas del infinitivo de los verbos. Las variaciones de las palabras de este diccionarios vienen determinadas por su conjugación: primera -ar, segunda -er y tercera -ir. Para cada palabra se formarán todos los tiempos verbales de los verbos regulares de su conjugación.

4.4.4. Creación de diagramas

Para representar las emociones detectadas en los textos de las noticias y de los comentarios se realizarán una serie de diagramas. Los métodos encargados de realizar esta tarea serán creaDiagramaTartaNoticia(), creaDiagramaTartaComentarios() y creaDiagramaLineasEmociones(). Estos métodos crearán todos los diagramas necesarios para una noticia específica.

Cada método tiene como parámetro de entrada un objeto de tipo Noticia del que se extraerá toda la información necesaria. El objeto Noticia contiene una lista con todos los comentarios de esa noticia y tanto los objetos Noticia como los objetos Comentario tienen un objeto de tipo Emoción que almacena toda la información del análisis emocional.

Los diagramas tendrán como título el titular de la noticia y serán guardados en disco con el mismo nombre y una descripción breve del tipo de diagrama. Debido a que el sistema de ficheros

de los sistemas operativos no soporta algunos caracteres se eliminarán del titular para evitar fallos al guardar los diagramas. También la longitud del nombre de los ficheros está limitada de modo que si supera la longitud el titular será recortado.

Cada tipo de emoción representada en los diagramas tendrá asociada un color específico: azul para tristeza (sadness), amarillo para miedo (fear), rojo para enfado (anger) y verde para alegría (joy).

4.4.4.1. Método creaDiagramaTartaNoticia()

El método creaDiagramaTartaNoticia crea un diagrama de secciones o tarta. Las secciones representan el porcentaje de palabras encontradas en el texto de la noticia de cada emoción. Se mostrará el número de palabras encontradas y el porcentaje asociado a cada sección del diagrama. Una leyenda indicará qué colores tiene cada emoción en la parte inferior del diagrama.

4.4.4.2. Método creaDiagramaTartaComentarios()

El método creaDiagramaTartaComentarios es muy similar al método creaDiagramaNoticia. La diferencia principal es que en vez de obtener el número de emociones encontradas en el texto de la noticia se obtiene el número de emociones encontradas entre todos los comentarios de la noticia.

Este método creará dos diagramas. El primer diagrama muestra el porcentaje de cada emoción de todos los comentarios en conjunto. El segundo además tendrá en cuenta los votos recibidos para cada comentario. Si un comentario tiene más votos negativos que positivos el número de emociones encontradas no se tiene en cuenta. En caso contrario esa diferencia positiva será un multiplicador para cada emoción encontrada en el comentario.

La finalidad de esta modificación es dar más importancia a las emociones de los comentarios que hayan sido apoyados por el resto de las personas y por contra penalizar las emociones de los comentarios que no sean apoyados por la gente. Esto puede ser un indicador más ajustado al sentir general de las personas sobre ésa noticia.

4.4.4.3. Método creaDiagramaLineasEmociones()

Por último creaDiagramaLineasEmociones creará dos gráficas más. La primera mostrará el número de comentarios de una noticia a lo largo del tiempo. Los datos serán representados por un conjunto de dos dimensiones en un diagrama de líneas.

El eje de las abscisas representa los intervalos de tiempo en los que se muestran los datos. En este caso los intervalos serán de una hora. Por un lado el eje de las ordenadas mostrará el número de comentarios en cada intervalo de tiempo. Por otro además mostrará el número de comentarios acumulado entre los intervalos de tiempo anteriores.

El segundo diagrama es similar y mostrará el número de palabras detectadas de cada emoción en los comentarios de una noticia a lo largo del tiempo. En este caso cada emoción será representada mediante una línea de color.

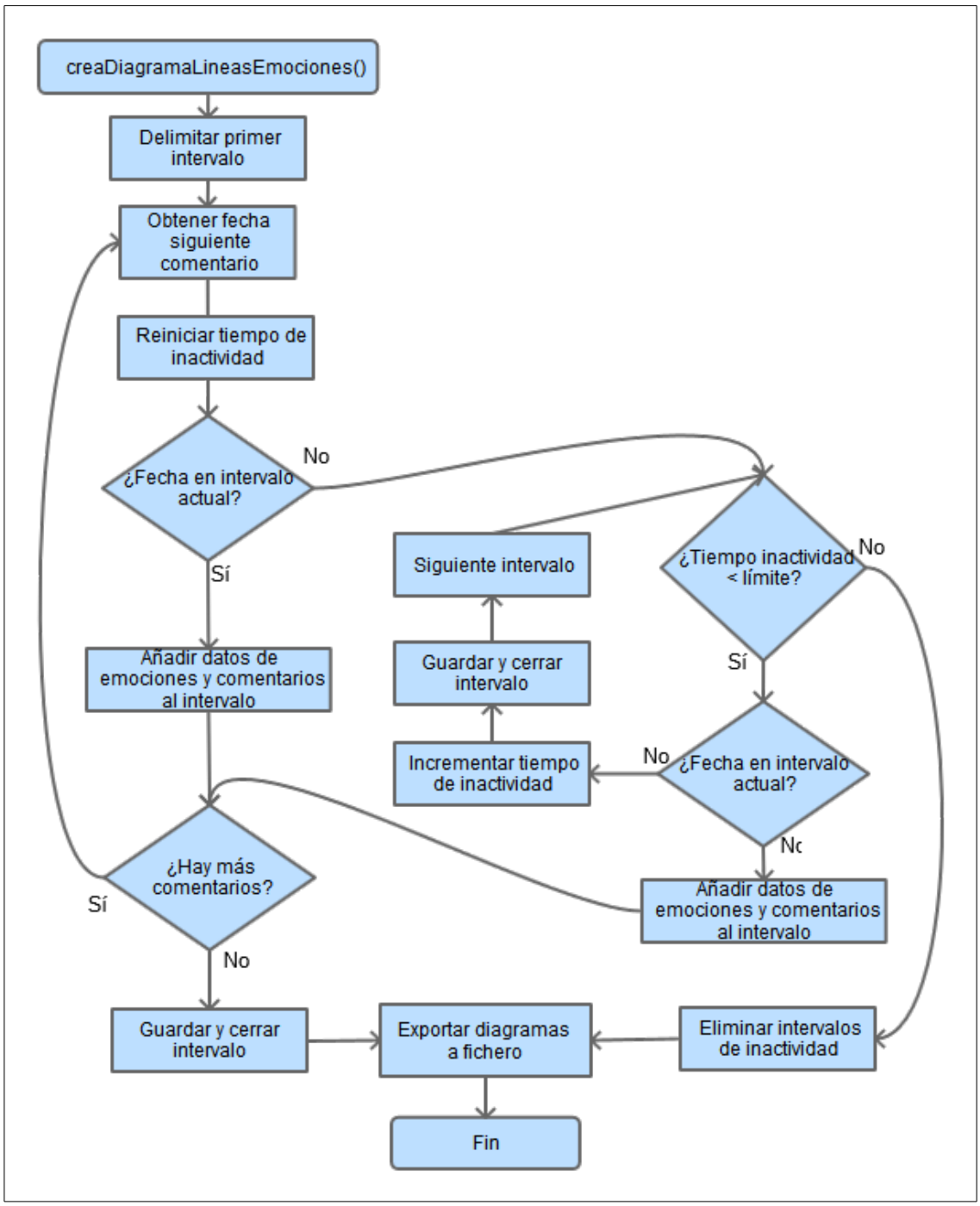

Figura 13. Diagrama de flujo del método creaDiagramaLineasEmociones()

5.Desarrollo

En este apartado se explicarán los detalles de la implementación del sistema tal y como ha sido diseñado.

Para implementar los módulos del sistema se han realizado dos proyectos en el entorno de programación Netbeans, resultando en dos aplicaciones de Java.

5.1.Módulo de extracción

La aplicación del módulo de extracción está compuesta de la clase principal encargada de la ejecución y de las clases que conforman la estructura de datos del sistema, como se mencionaron en el apartado 4.2 de este documento. La aplicación tiene incorporada además las librerías Jsoup, encargada de descargar y analizar el código HTML de la noticia, XStream que realiza las tareas de almacenamiento en ficheros y recuperación de información y la librería JFreeChart que se encarga de crear los diagramas que representan los datos del análisis emocional de las noticias.

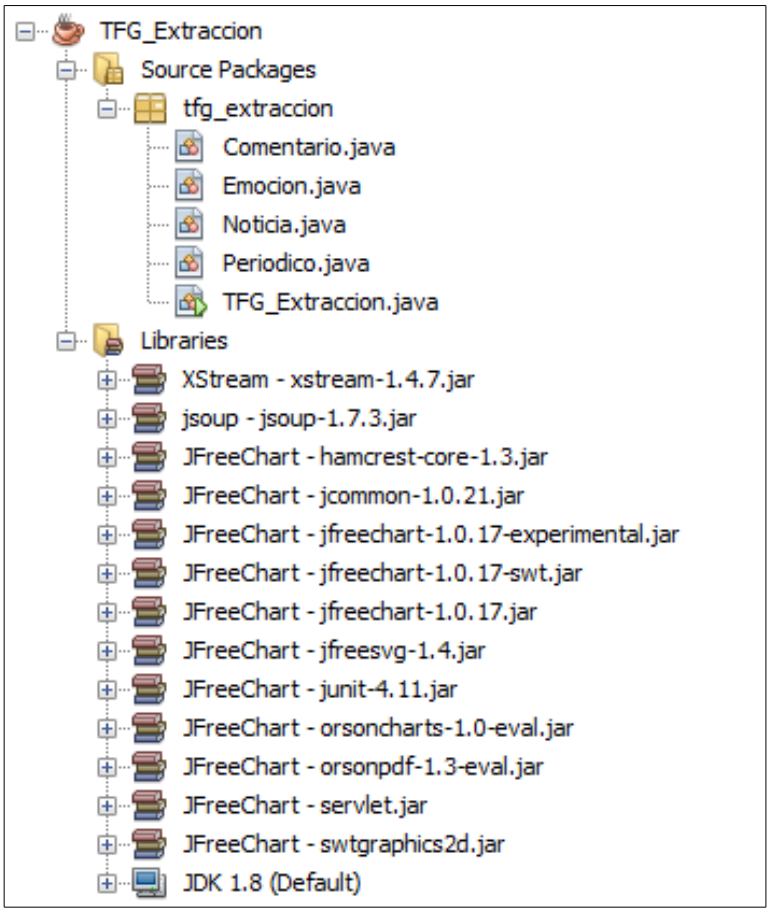

Figura 14. Estructura del proyecto del módulo de extracción.

Como se ha detallado en la parte de diseño, toda la información se almacena en una lista de objetos de tipo Periódico. Esta lista se encuentra en la clase principal para que se pueda operar fácilmente con ella.

También en la clase principal existe una variable que indica si se desea actualizar noticias existentes. En caso afirmativo se lee un fichero en el directorio raíz del proyecto o de la aplicación con el nombre "noticias.xml". Si este fichero no existe, o bien, es un fichero no generado por el módulo de extracción se mostrará un error y la aplicación se cerrará. Si el fichero es correcto, se almacenará de nuevo toda la información en la lista de periódicos.

Para la obtención de noticias se han realizado tres métodos en la clase Periódico que obtienen las URL con la herramienta Jsoup y crean los objetos de tipo Noticia y los añaden al objeto Periódico pertinente: fetchNoticiasFromRSS(), fetchNoticiasFromLoMas() y fetchNoticiasFromURL(). Se han añadido otras dos funciones auxiliares a esta clase que son isNoticiaInPeriodico(), que comprueba si la noticia que se está obteniendo está previamente guardada y fixRSSURL(), que arregla la dirección de la noticia obtenida a partir de los RSS del periódico.

En cuanto a la obtención de datos de las noticias se ha realizado un método llamado fetchDatosNoticia() en la clase Noticia según el diseño anterior. Para la extracción de datos se ha utilizado la herramienta Jsoup. Este método es llamado para cada noticia de cada periódico.

Para la obtención de comentarios se ha realizado el método fetchComentariosNoticia() siguiendo las líneas del diseño. Al igual que fetchDatosNoticia(), se utiliza Jsoup para obtener los datos y el método es llamado por todas las noticias de cada periódico.

Algunas noticias no contienen texto y no son compatibles. Para eliminar estas noticias será necesario comprobar una a una si tiene o no el texto de la noticia y borrar las que no lo tengan. Para borrar de la lista correctamente hay que iterar sobre el tamaño en cada momento para evitar acceder a una posición de memoria no permitida. Así, mientras el índice sea menor que el tamaño total, se comprueba que tenga el texto o no. En caso afirmativo el índice se incrementará para comprobar en la siguiente iteración la siguiente noticia. En caso contrario se eliminará la noticia que ocupa el lugar del índice actual, pero en este caso no se incrementará el índice ya que una vez se borra un elemento el elemento siguiente pasa a ocupar su lugar.

Para almacenar los datos de forma permanente en un fichero se ha utilizado XStream. Esta herramienta traduce un objeto o, en este caso, una lista de objetos en texto en formato XML. Para objetos que tengan referencias a otros utiliza un sistema de referencia por niveles del árbol XML. Esto permite que cuando se lea de nuevo el fichero y se convierta de nuevo a una lista de objetos los datos permanezcan intactos. Para facilitar la inspección de los datos se ha agregado al inicio del fichero una cabecera en la primera línea que deberá ser eliminada en la lectura del fichero para poder recuperar los datos.

5.2. Módulo de análisis emocional

La aplicación del módulo de análisis emocional mantiene las clases de estructura de datos y las librerías incorporadas que tiene el módulo de extracción y se diferencia de éste en la clase principal, que realiza otras tareas.

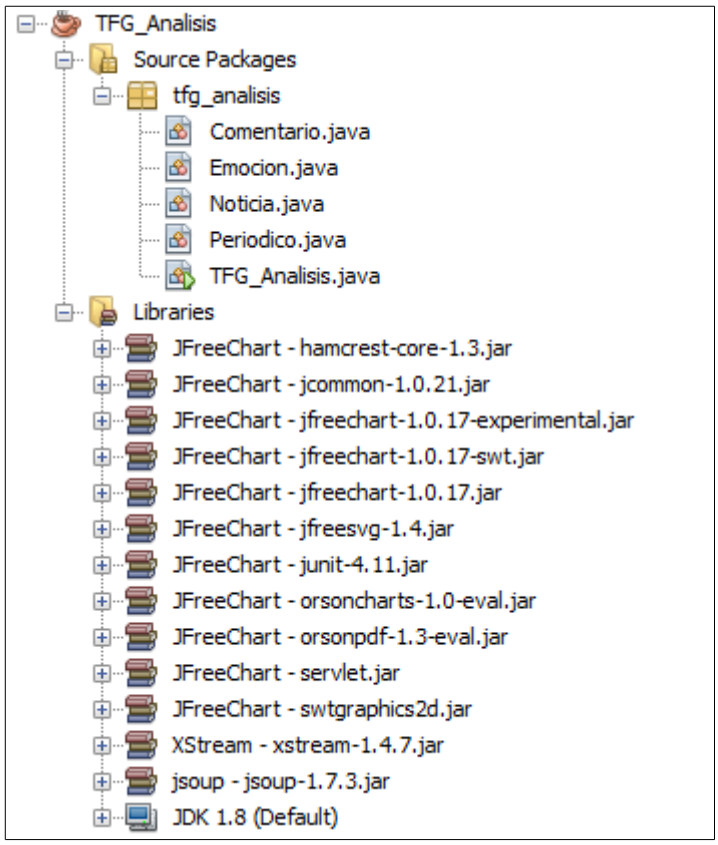

Figura 15. Estructura del proyecto del módulo de análisis emocional.

Al igual que el módulo de extracción, este módulo incorpora una lista de objetos de tipo Periódico en la que guardar todos los datos.

Para recuperar los datos obtenidos anteriormente se leerá de un fichero en el directorio raíz llamado “noticias.xml”. Si este fichero no existe o no está creado por el módulo de extracción se mostrará un aviso de error de lectura y el programa terminará su ejecución.

Para la lectura correcta de los diccionarios de emociones han de estar en una carpeta llamada “diccionarios” bajo la carpeta raíz del proyecto o aplicación. El prototipo de fichero es “[emoción]_stem_[tipo].txt” siendo emoción “joy”, “anger”, “fear” o “sadness” y el tipo “TAL-CUAL”, “sufijos” o “VERBOS”. Si no se encontrara alguno de los diccionarios se mostrará un aviso de error de lectura y el programa terminará su ejecución.

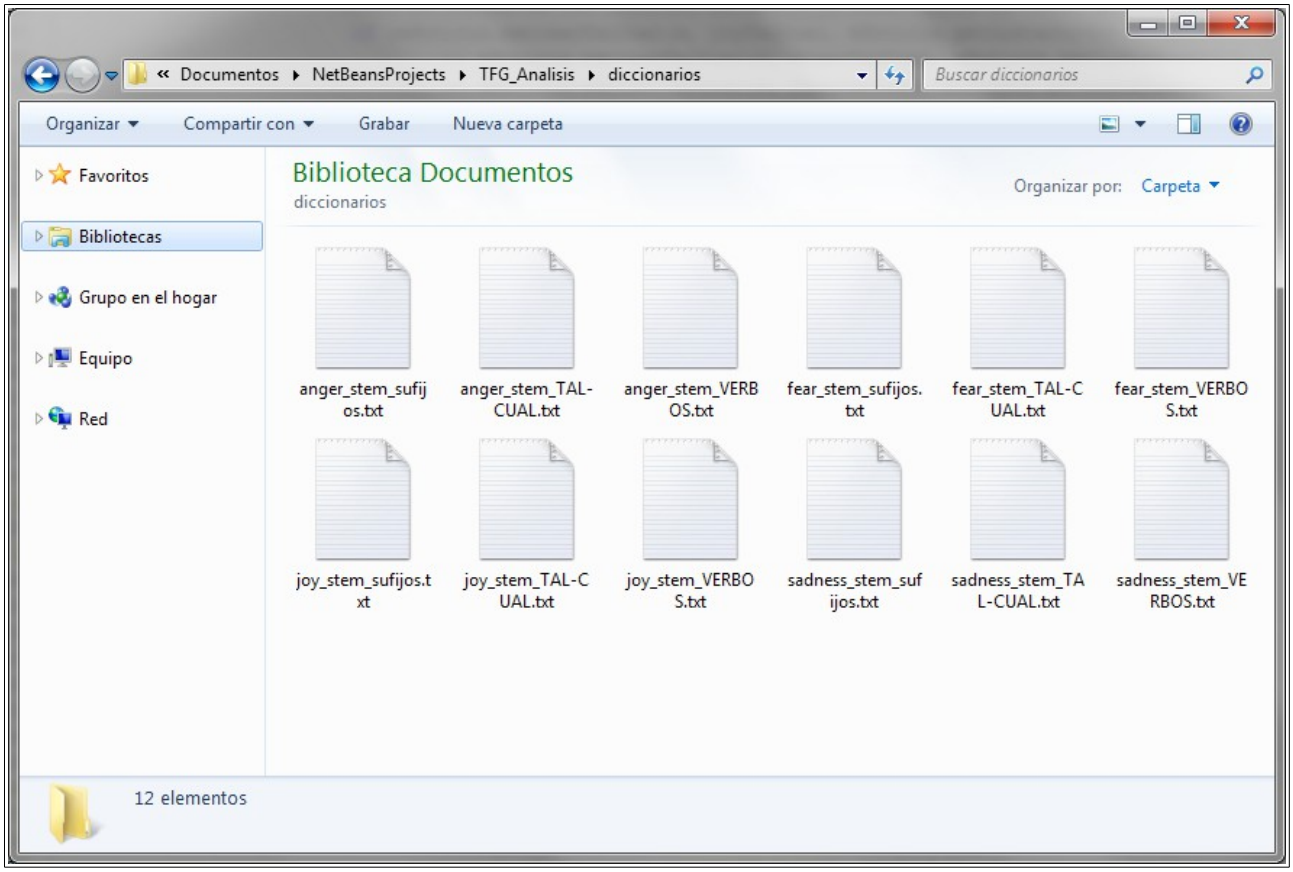

Figura 16. Diccionarios de palabras de emociones.

Cada objeto de tipo Noticia y de tipo Comentario contienen un objeto de la clase Emoción. Esta clase contiene los métodos que realizan el análisis emocional (ver apartado 4.4.3).

Tras realizar el análisis de emociones, se crean los diagramas. Para los diagramas de tarta se necesita un conjunto de datos en el que se añade un valor determinado por tipo de emoción. En el caso del diagrama para el texto de las noticias, se trata del número de emociones encontradas en la noticia. Para el diagrama de comentarios se trata del número de emociones encontradas entre todos los comentarios.

La figura 17 muestra un diagrama de tarta. En él se observa el número de emociones encontradas y la proporción de cada una de ellas.

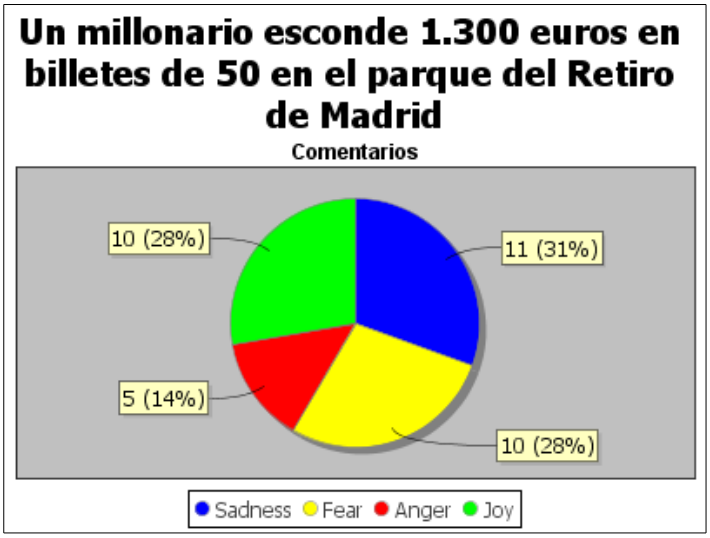

Figura 17. Ejemplo de diagrama de tarta.

Los diagramas de líneas se componen de series de datos que representan las líneas del diagrama. Cada dato de la serie se compone de dos valores: el número del intervalo de la hora y el valor asociado. La figura 18 muestra un diagrama de este tipo, donde se puede ver la evolución de los comentarios a lo largo de las horas tras la publicación del primer comentario.

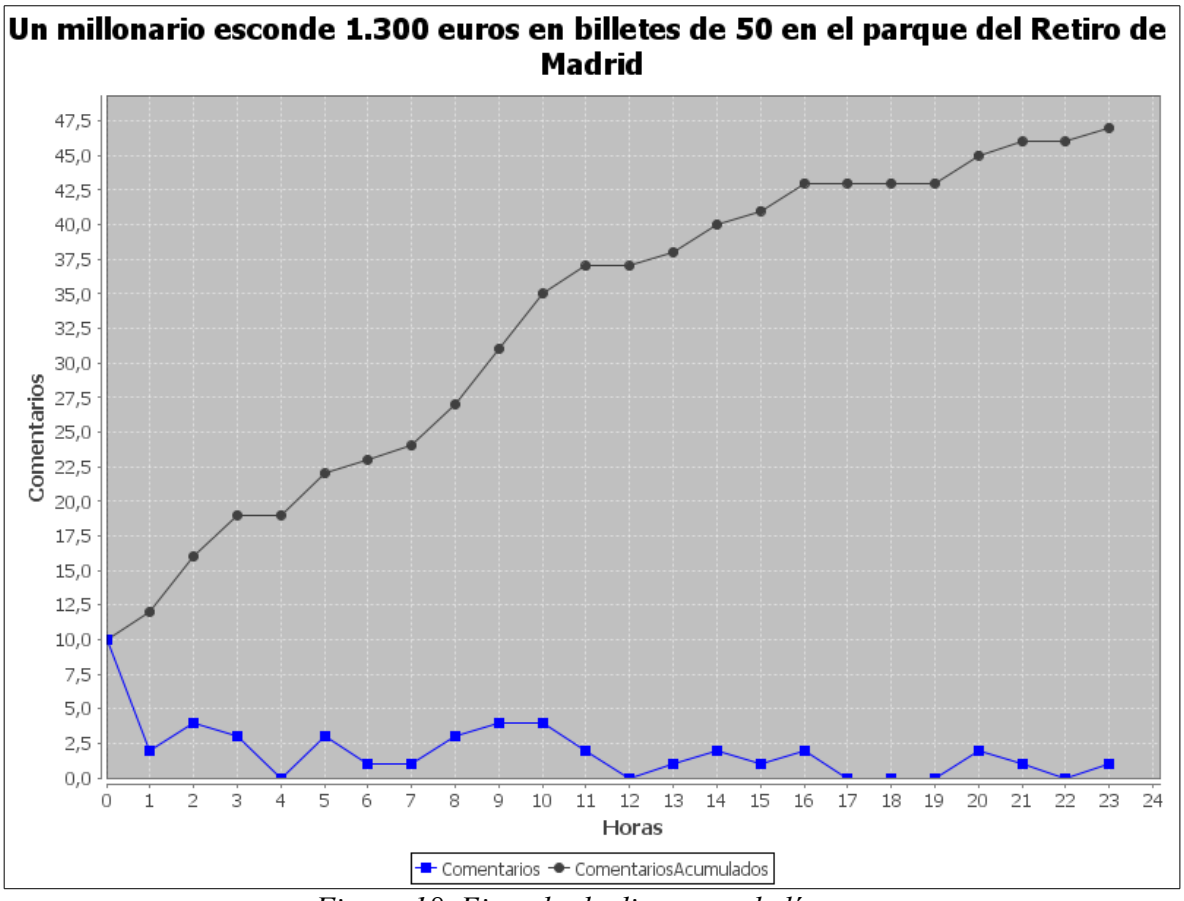

Figura 18. Ejemplo de diagrama de líneas.

En la serie del diagrama de líneas de comentarios este valor es el número de comentarios por cada intervalo para una serie y el número de comentarios acumulado para la otra. En el ejemplo se pueden observar los valores obtenidos para cada intervalo. En el diagrama de líneas de emociones cada tipo emoción tendrá una serie y el valor asociado es el número de emociones detectadas en cada intervalo.

Finalmente,estos diagramas se guardarán bajo una carpeta llamada “diagramas”. Los nombres siguen el siguiente prototipo “[TituloNoticia]-[TipoDiagrama].png” pudiendo ser “TipoDiagrama” los siguientes tipos “TartaComentarios”, “TartaComentariosPotenciados”, “TartaNoticia”, “LineasComentarios”, “LineasEmociones”.

6.Ejemplos de análisis de noticias

Para ilustrar el funcionamiento del sistema desarrollado se ha escogido como noticia de ejemplo una de las más comentadas en las últimas semanas, el anuncio por parte del presidente del gobierno de la abdicación del rey de España.

Figura 19. Ejemplo de noticia.

Enlace permanente a la noticia: http://www.20minutos.es/noticia/2155465/0/mariano-rajoy/anuncia-abdicacion-abdica/rey-juan-carlos/

6.1. Análisis de la noticia

6.1.1. Extracción de la información de la noticia

Lo primero que realiza el módulo de extracción es crear el objeto que representa el periódico, en este caso 20minutos.es, a partir del cual obtener las noticias. Como ya se tiene la URL de la noticia se utiliza el método del periódico fetchNoticiaFromURL, el cual crea un objeto en el periódico con la URL de la noticia. Acto seguido se adquieren los datos de la noticia con el método llamado fetchDatosNoticia. La siguiente imagen muestra los datos obtenidos a excepción del texto de la noticia que por motivos de espacio no se muestra. Además se crea el objeto emoción a partir del texto de la noticia creando una lista de palabras preparadas para ser analizadas.

```
Periodico:
 Nombre: 20 Minutos
 URL: http://www.20minutos.es/
 RSS: http://20minutos.feedsportal.com/c/32489/f/478284/index.rss

Ejecución de fetchNoticiaFromURL()
Noticia:
 URL: http://www.20minutos.es/noticia/2155465/0/mariano-rajoy/anuncia-abdicacion-abdica/rey-juan-carlos/
 Titular: null
 Fecha: null
 Autor: null
 Votos positivos: 0
 Votos negativos: 0
 Periodico: 20 Minutos

Ejecución de fetchDatosNoticia()
Noticia:
 URL: http://www.20minutos.es/noticia/2155465/0/mariano-rajoy/anuncia-abdicacion-abdica/rey-juan-carlos/
 Titular: Mariano Rajoy anuncia que el rey Juan Carlos I abdica
 Fecha: 2014/06/02 10:01:00
 Autor: ISRA ÁLVAREZ
 Votos positivos: 2
 Votos negativos: 5
 Periodico: 20 Minutos
```

Figura 20. Información obtenida de una noticia.

A continuación se procede a obtener los comentarios de la noticia mediante el método fetchComentariosNoticia. El siguiente comentario muestra a modo de ejemplo la información almacenada para cada comentario de la noticia. Al igual que con el texto de las noticias, para cada comentario se crea un objeto emoción a partir del texto del comentario y se crea una lista de palabras en las que se eliminan caracteres extraños para poder ser analizado más adelante.

```
Comentario:
 Texto: a mi me es indiferente lo que suceda en este tema
 Fecha: 2014/06/02 13:39
 Autor: chocosonrisas
 Votos positivos: 0
 Votos negativos: 0
 Noticia: Mariano Rajoy anuncia que el rey Juan Carlos I abdica
```

Figura 21. Información obtenida de un comentario.

Llegados a este punto la herramienta XStream se encarga de guardar toda la información relativa a los periódicos, noticias, comentarios y sus respectivas emociones en un fichero llamado “noticias.xml”.

6.1.2. Análisis emocional de la noticia

Para poder ejecutar el módulo de análisis es necesario copiar el fichero “noticias.xml” guardado a partir del módulo de extracción en la carpeta raíz del ejecutable. Una vez el fichero es leído se tienen los mismos datos de las noticias al ser guardadas.

Si el fichero ha sido leído correctamente se procede a leer los diccionarios de emociones. Estos diccionarios deben estar en una carpeta llamada “diccionarios”. Tras su lectura se almacena cada diccionario en listas de palabras.

Cuando todos los diccionarios se han leído correctamente se procede a realizar el análisis emocional de la noticia y después todos los comentarios de cada noticia. Al analizar el texto de esta noticia se han encontrado las siguientes palabras.

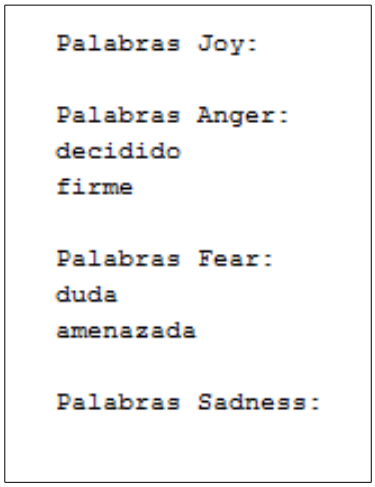

Figura 22. Emociones encontradas en una noticia.

Cada comentario es también analizado de manera separada. La siguiente imagen es una muestra de las palabras encontradas entre todos los comentarios.

Palabras Joy:	Palabras Anger:	Palabras Fear:	Palabras Sadness:
buena	miedo	esfuerzos	durado
disfrute	miedo	malo	perdona
gustan	moleste	problemas	lastima
buen	decidir	graves	pobre
viva	miedo	crisis	verguenza
alegria	preocupe	dudas	perdon
viva	decidira	mal	negativo
encantado	cargando	graves	perdon
viva	loca	contrario	perdon
viva	animo	malo	verguenza
gusto	rabia	tema	pobre
viva	valor	duda	alteracion
viva	miedo	temo	pobres
viva	molestado	tema	sentimiento
viva	miedo	malo	criticables
viva	protestas	tema	escandalosa
viva	decidir	triste	pobre
viva	decidieron	pena	decae
buen	miedo	pesar	dure
gustaria	ofender	consciente	perdon
viva	miedo	amenaza	descontento
viva	preocupes	daño	
buen	preocupes	mal	
viva	miedo	mala	
alegria	preocupes	mal	
viva	decidir	conscientes	
bueno	protestar	consciente	
celebrar	protestar	mala	
aprecio	loco	tema	
gustaria	cargo	interesado	
buena	ira	sufrio	
gustaria	decidan	problemas	

Figura 23. Extracto de emociones encontradas en todos los comentarios de una noticia.

Utilizando los datos del análisis se construyen una serie de gráficas que representan de una forma más manejable los datos facilitando el posterior estudio.

La primera gráfica se trata de un diagrama de tarta que muestra la proporción de emociones encontradas en el texto de la noticia. Como se puede observar, esta noticia es bastante neutra, en el sentido de que prácticamente no se utilizan palabras asociadas con emociones. Tan sólo se detectaron dos palabras representando miedo y otras dos representando ira (50% cada emoción).

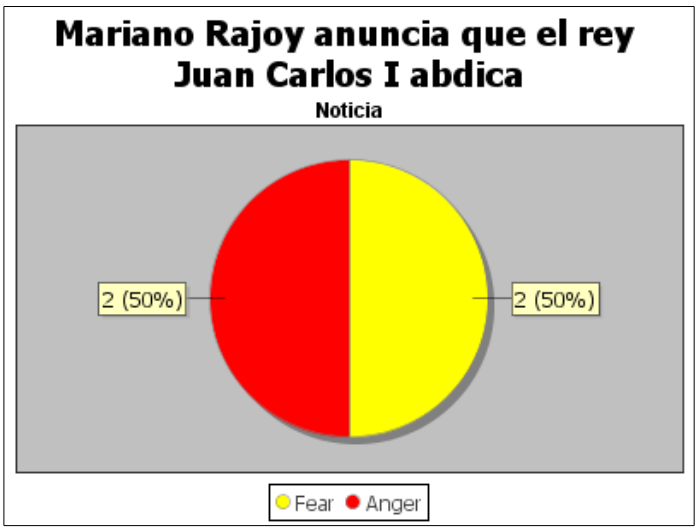

Figura 24. Diagrama de tarta de las emociones encontradas en una noticia.

La segunda gráfica es similar a la anterior pero en este caso muestra las emociones encontradas entre todos los comentarios. Como se puede observar, en los comentarios de los usuarios sí se detectan bastantes palabras asociadas con emociones. En este caso concreto se detectó un 9% de tristeza, un 33% de miedo, un 20% de ira y un 38% de alegría.

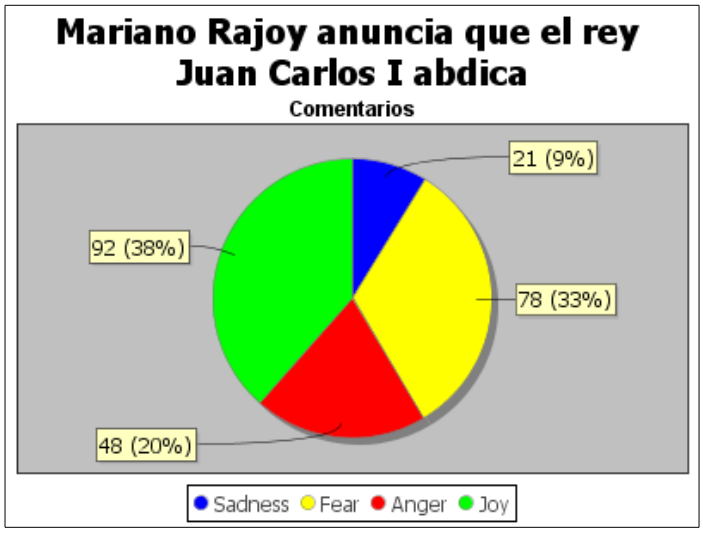

Figura 25. Diagrama de tarta de emociones encontradas en los comentarios de una noticia.

La siguiente gráfica se basa en los mismos datos que la anterior, pero en este caso las emociones de los comentarios con votos negativos no se han incluido y las de los comentarios con votos positivos se han multiplicado por la diferencia de votos positivos y negativos que tienen. El objetivo en este caso es mostrar las emociones transmitidas por todos aquellos comentarios que no han sido valorados negativamente por otros usuarios. En este caso se observa que los porcentajes se modifican ligeramente. Aun no sufriendo grandes variaciones, y siendo fear y joy las emociones predominantes, antes se detectaba un mayor nivel de joy que de fear y ahora sucede lo contrario. En cualquier caso, siguen manteniéndose dichas emociones como las predominantes frente a sadness o anger.

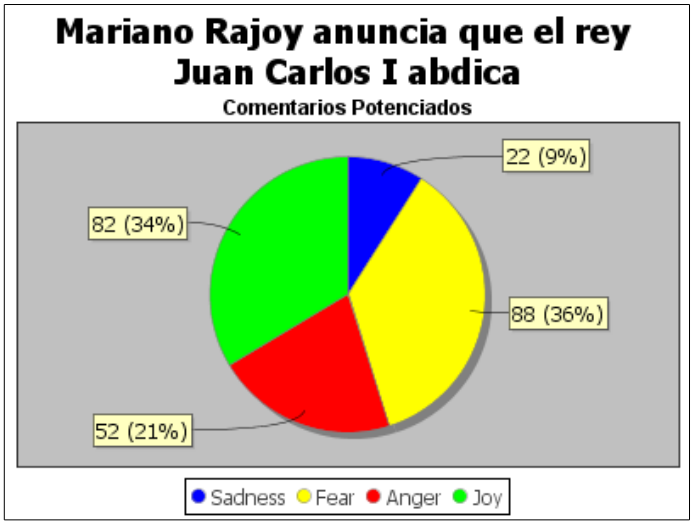

Figura 26. Diagrama de tarta de emociones encontradas en comentarios potenciados.

El siguiente diagrama permite observar la evolución del número de comentarios a lo largo del tiempo y muestra a su vez la cantidad de comentarios acumulados. Como se puede observar, la mayoría de los comentarios se produjeron durante las primeras horas tras las cuales sólo se publicaron un porcentaje muy pequeño de comentarios.

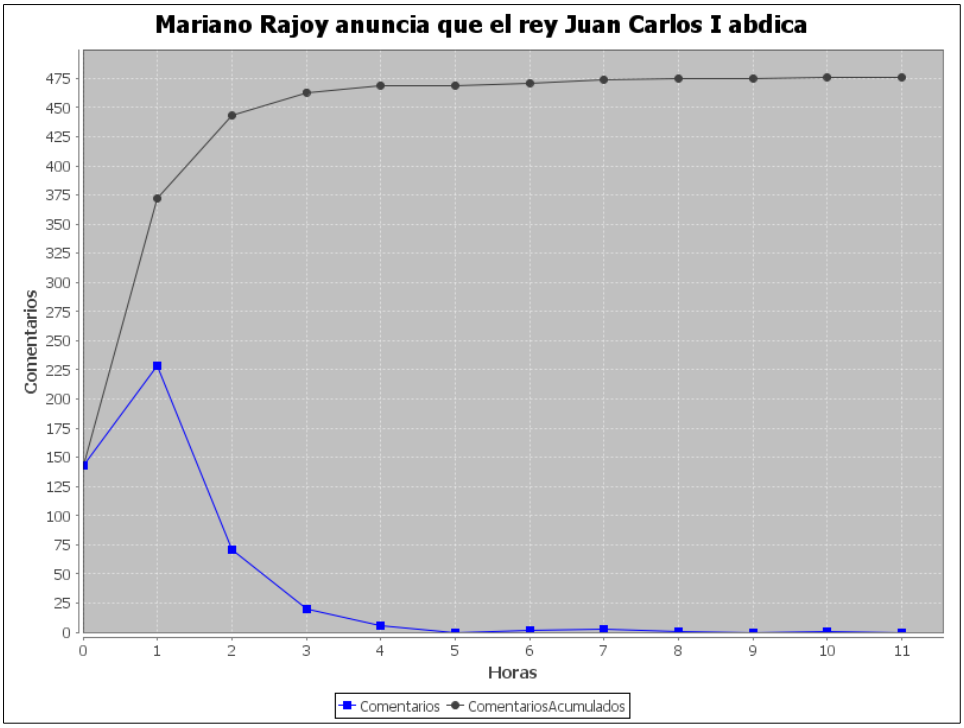

Figura 27. Diagrama de líneas del número de comentarios a lo largo del tiempo.

Por último, el siguiente diagrama muestra el análisis emocional de los comentarios en el tiempo. En cada intervalo, de una hora de duración, los valores mostrados son la suma de las emociones de cada comentario

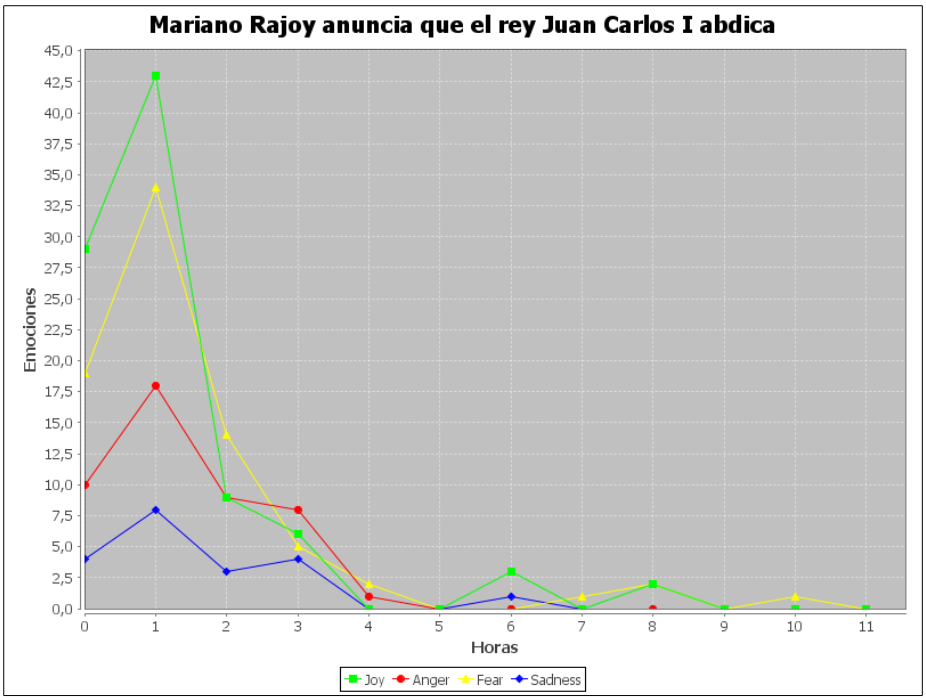

Figura 28. Diagrama de líneas del número de emociones encontradas a lo largo del tiempo.

6.2. Análisis de noticias relacionadas

Para comparar las emociones plasmadas en varias noticias relacionadas, se han escogido tres noticias relacionadas con la anterior. Entre todas las relacionadas se han elegido las que más comentarios tenían en ese momento con el fin de obtener una muestra lo más grande posible. Las noticias analizadas son las siguientes:

Noticia 1: "Mariano Rajoy anuncia que el rey Juan Carlos I abdica"

URL:http://www.20minutos.es/noticia/2155465/0/mariano-rajoy/anuncia-abdicacion-abdica/rey-juan-carlos/

Noticia 2: "El rey don Juan Carlos abdica: «Hoy merece pasar a primera línea una generación más joven»"

URL: http://www.20minutos.es/noticia/2155479/0/minuto-a-minuto/rey-juan-carlos/abdica/

Noticia 3: "El Congreso aprueba la ley de abdicación de don Juan Carlos con el 85% de los votos a favor"

URL: http://www.20minutos.es/noticia/2163899/0/votacion-ley-abdicacion/congreso/minuto-a-minuto/

Noticia 4: “Así fue la proclamación de Felipe VI como rey de España, contada en directo minuto a minuto”

URL: http://www.20minutos.es/noticia/2169826/0/proclamacion-rey-felipe-vi/abdicacion-juan-carlos/directo/

Para cada noticia se ha realizado el mismo proceso que en el ejemplo anterior. Se ha obtenido toda la información de cada noticia y sus respectivos comentarios. Después se ha realizado el análisis emocional a los textos de las noticias y comentarios. Los diagramas que aparecen a continuación son una representación de las emociones extraídas de cada noticia.

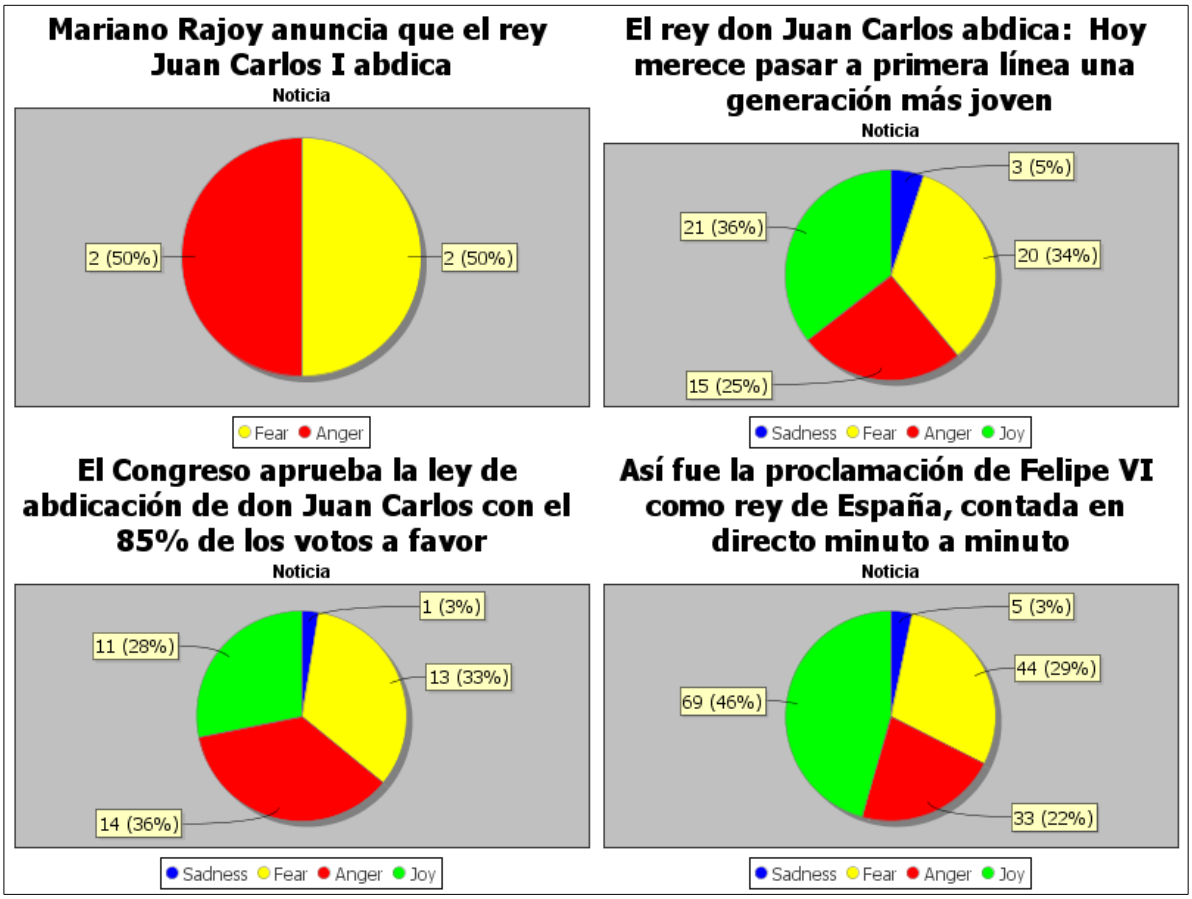

Figura 29. Diagrama de tartas de emociones de noticias relacionadas.

En esta primera imagen se observan las emociones detectadas en los textos de las noticias analizadas. Por lo que se puede observar, en la primera noticia solamente se han detectado cuatro palabras. Esto puede deberse a que el texto de la noticia es considerablemente corto comparado con el de las demás. En el resto de noticias se observa una proporción parecida de emociones detectadas entre ellas, con unos datos de media del 4% para las palabras de emociones de tristeza, un 32% de miedo, un 28% de ira y un 36% de alegría.

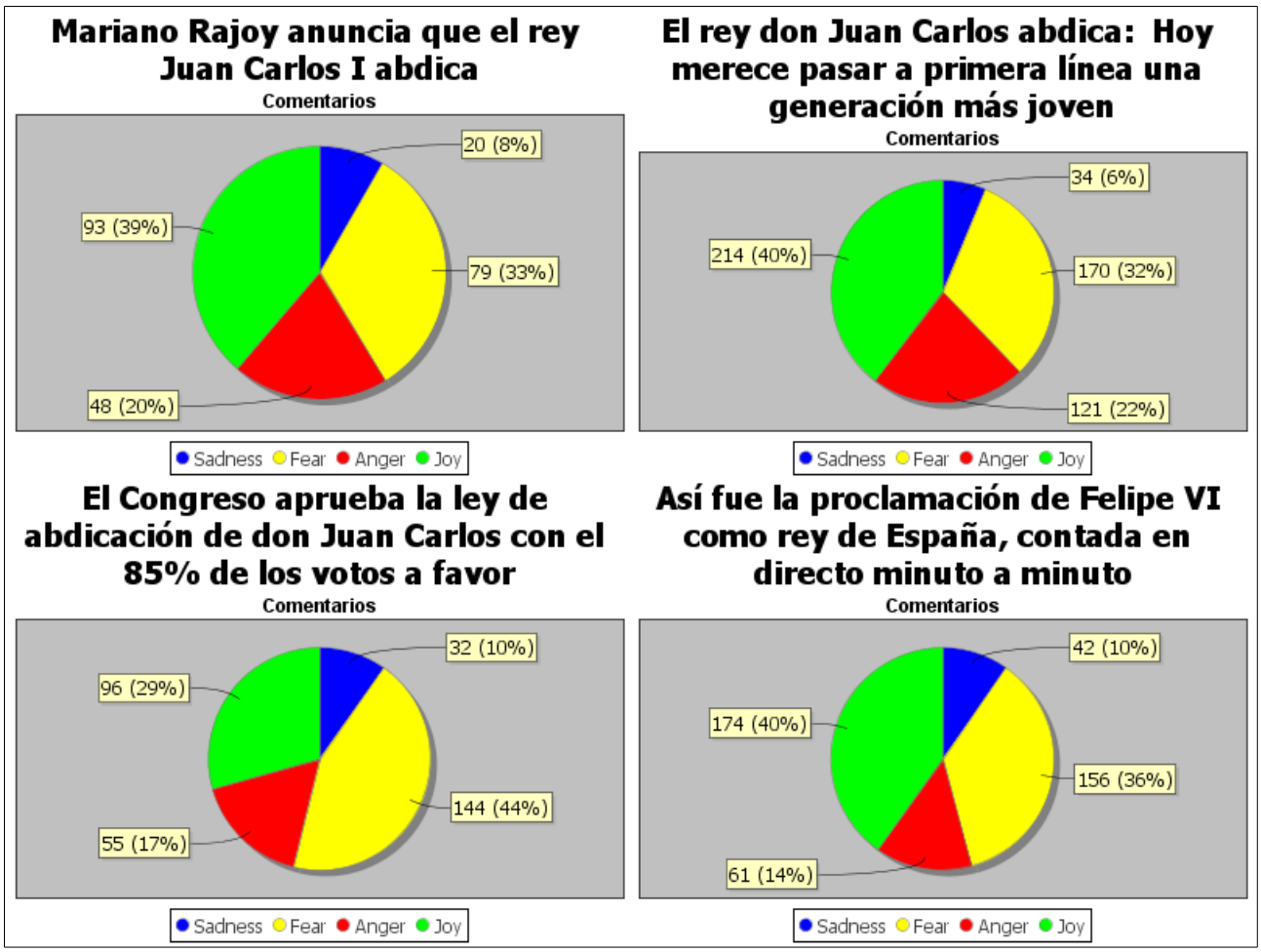

Figura 30. Diagrama de tarta de emociones de los comentarios de las noticias relacionadas.

En esta imagen se contemplan las palabras de emociones detectadas en el conjunto de los comentarios de cada noticia. Los datos mostrados en cada una son muy similares y parecen seguir en la línea de emociones encontradas en el texto de cada noticia, a excepción de la primera. Las medias de los porcentajes para cada emoción encontrada son un 8% de emociones de tristeza, un 37% de miedo, un 19% de ira y un 36% de alegría.

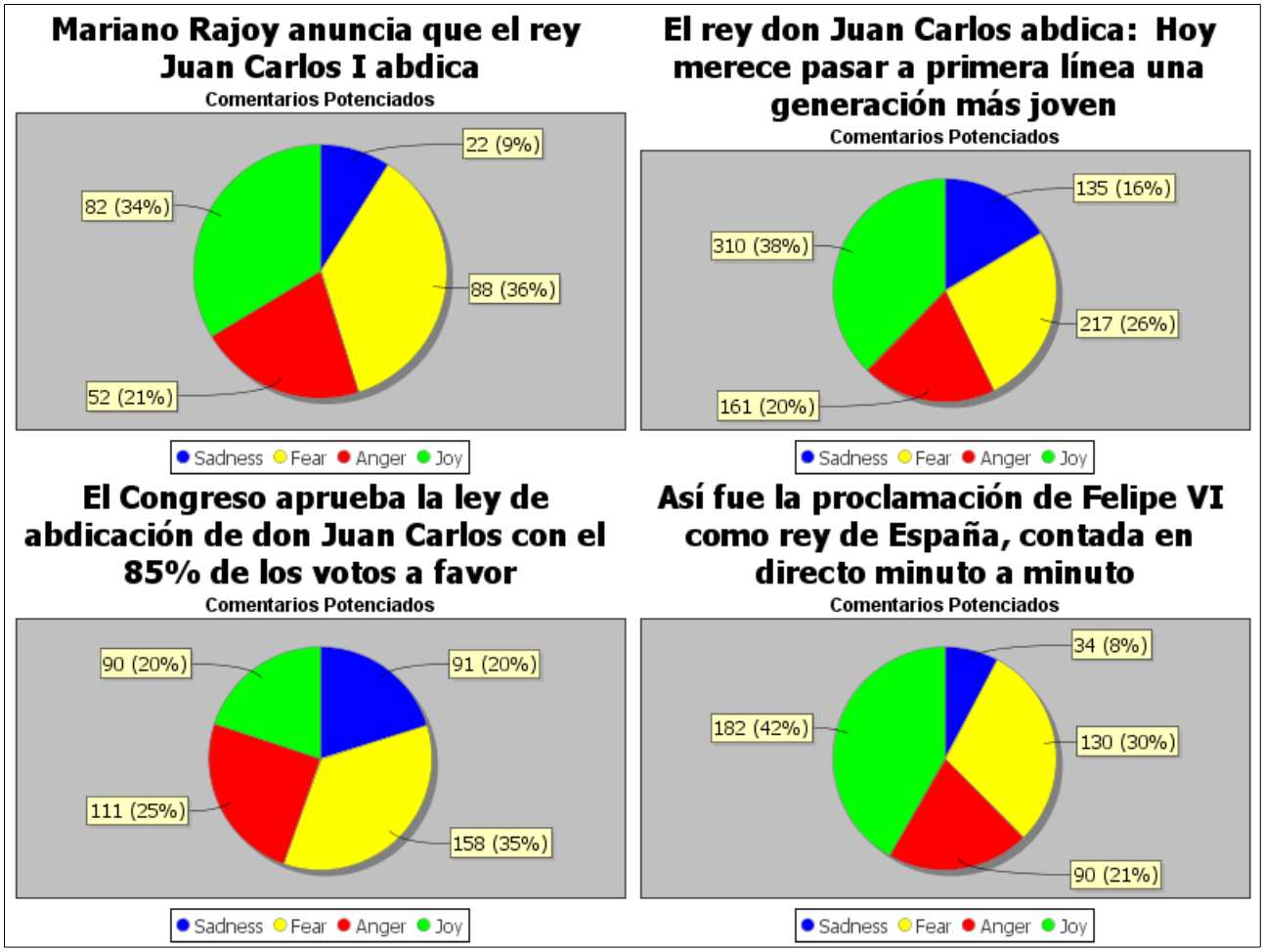

Figura 31. Diagrama de tarta de emociones de comentarios potenciados de noticias relacionadas.

El análisis de diversas noticias ha motivado la consideración sobre cómo se pueden valorar los comentarios, en función de las puntuaciones recibidas por cada comentario. En el anterior conjunto de gráficas los valores de las emociones detectadas reflejan la decisión tomada al respecto. Los comentarios que obtuvieron una valoración positiva han sido favorecidos multiplicando el número de emociones detectadas por el valor de la puntuación. Los comentarios que obtuvieron una valoración negativa no añaden el número de emociones detectadas al gráfico. En este caso la tendencia cambia ligeramente. La media de emociones encontradas de tristeza es del 13%, de miedo un 32%, de ira un 22% y de alegría un 33%. El cambio más significativo se da en la tercera noticia cuando las emociones que indican miedo pasan del 10% al 20% y las de ira del 17% al 25%. En la segunda noticia también aumenta las emociones encontradas para ira, del 6% al 16%. En el resto de noticias los porcentajes se mantienen más bien estables.

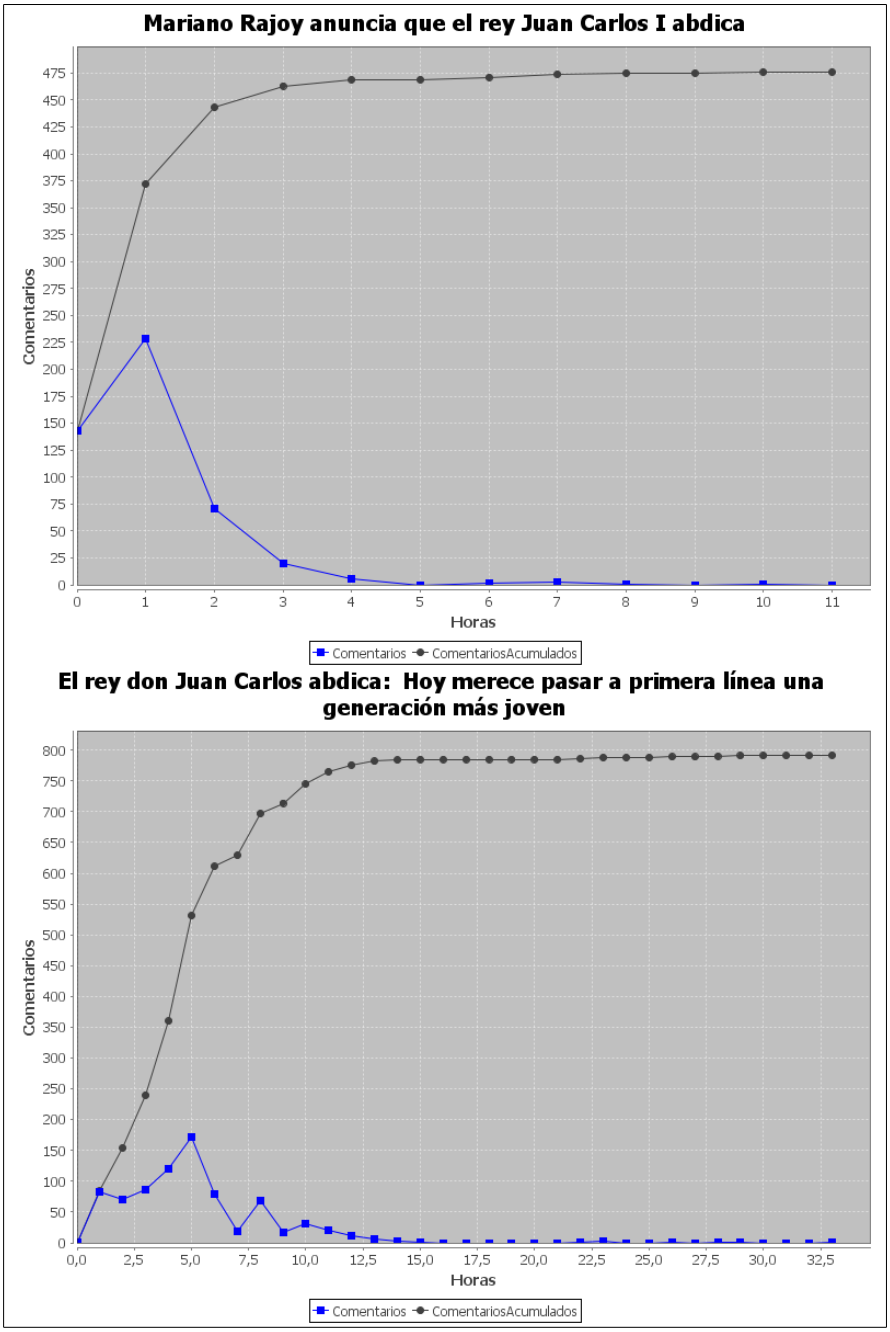

Figura 32. Diagrama de líneas de comentarios de noticias relacionadas. Parte 1.

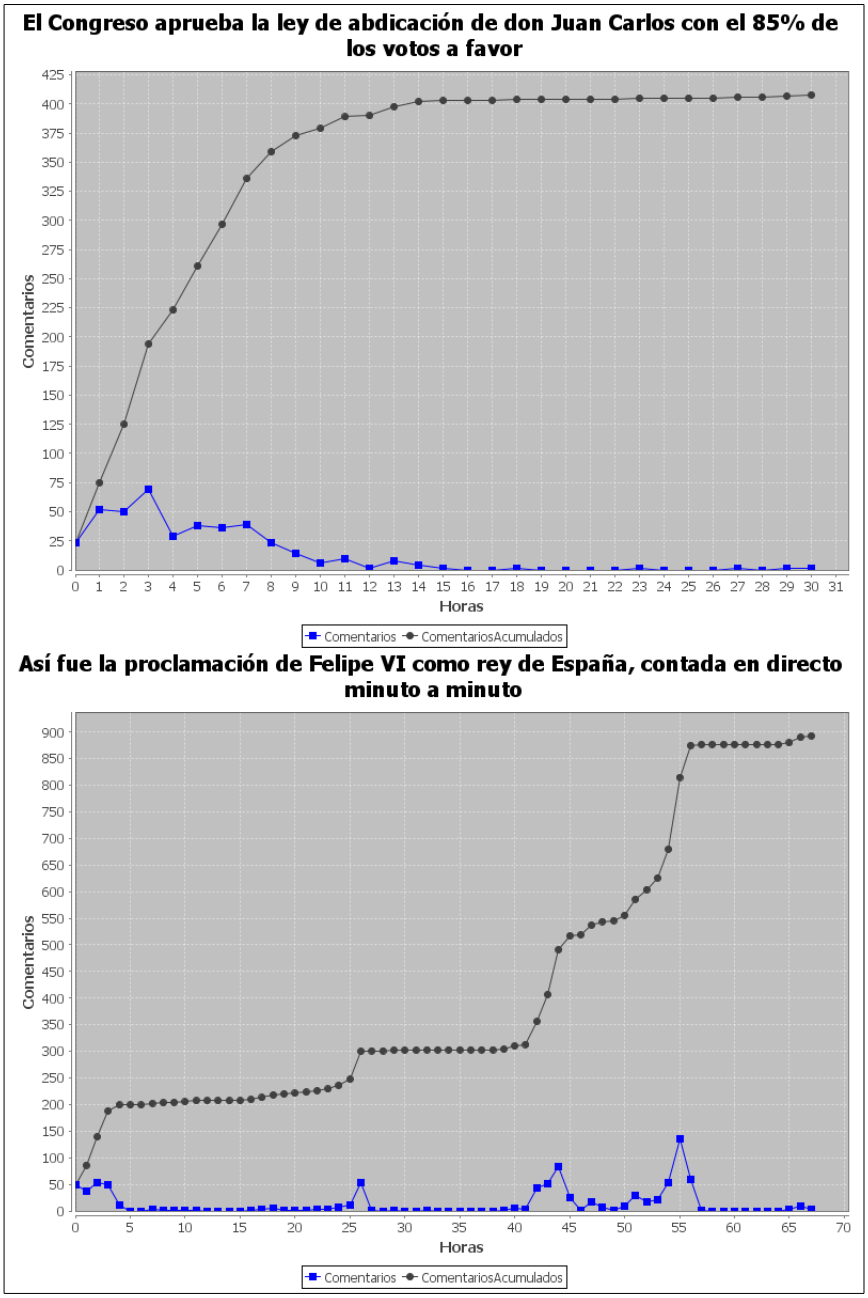

Figura 33. Diagrama de líneas de comentarios de noticias relacionadas. Parte 2.

En los siguientes diagramas (figuras 32 y 33) se muestra el número de comentarios a lo largo del tiempo. Las tres primeras noticias siguen un patrón similar en los comentarios acumulados. Las tres obtienen el 90% de los comentarios en las primeras horas desde la publicación de la noticia pasando a recibir el resto de comentarios de manera muy pausada. En cambio, la cuarta noticia ha sido actualizada minuto a minuto, narrando la noticia según han ido ocurriendo los acontecimientos de la noticia a lo largo de tres días, por eso los comentarios acumulados aparecen de manera escalonada. En la parte final, se observa un pico aproximadamente a las doce del mediodía del 19 de junio y otro pico final a las once de la noche.

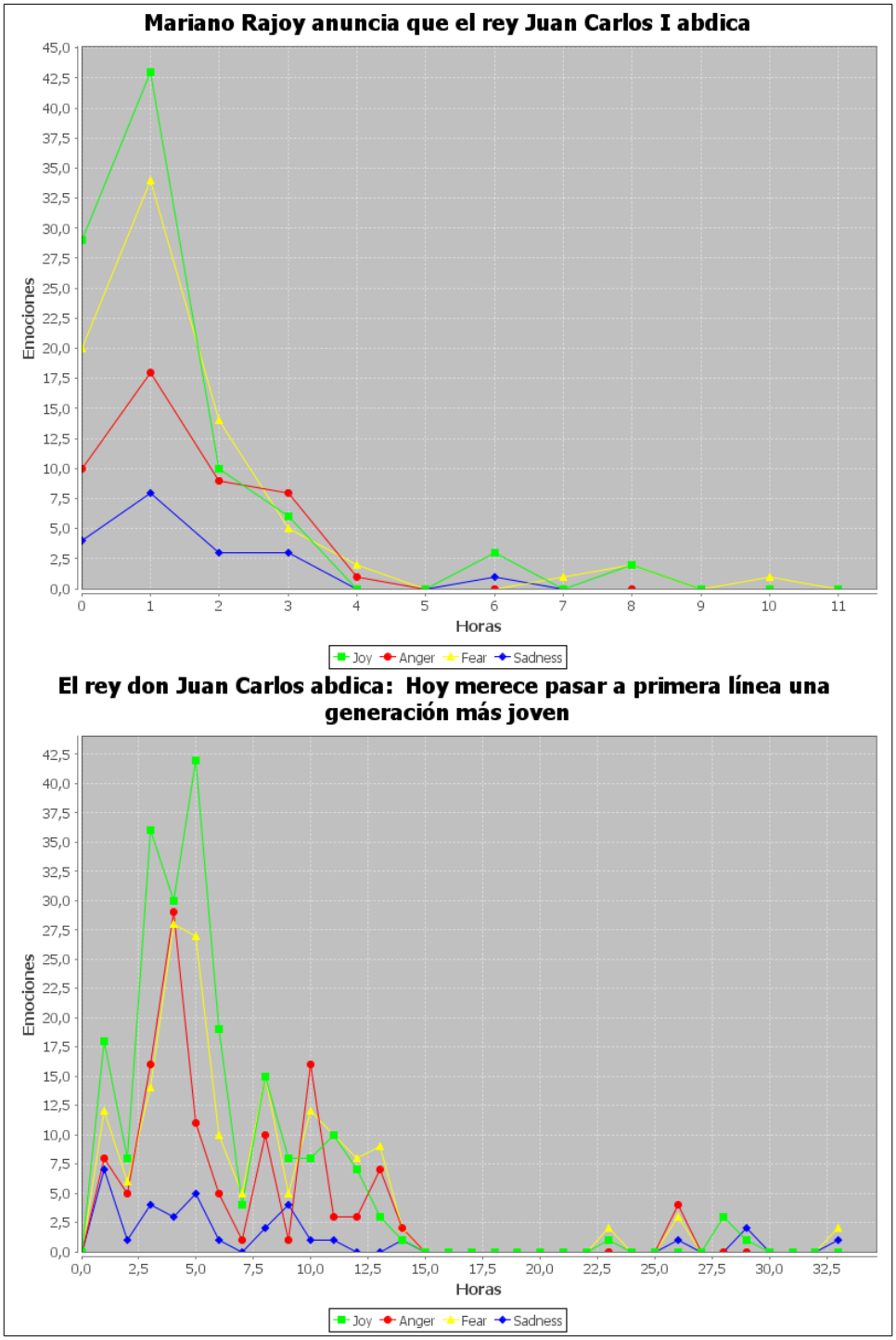

Figura 34. Diagrama de líneas de emociones de noticias relacionadas. Parte 1.

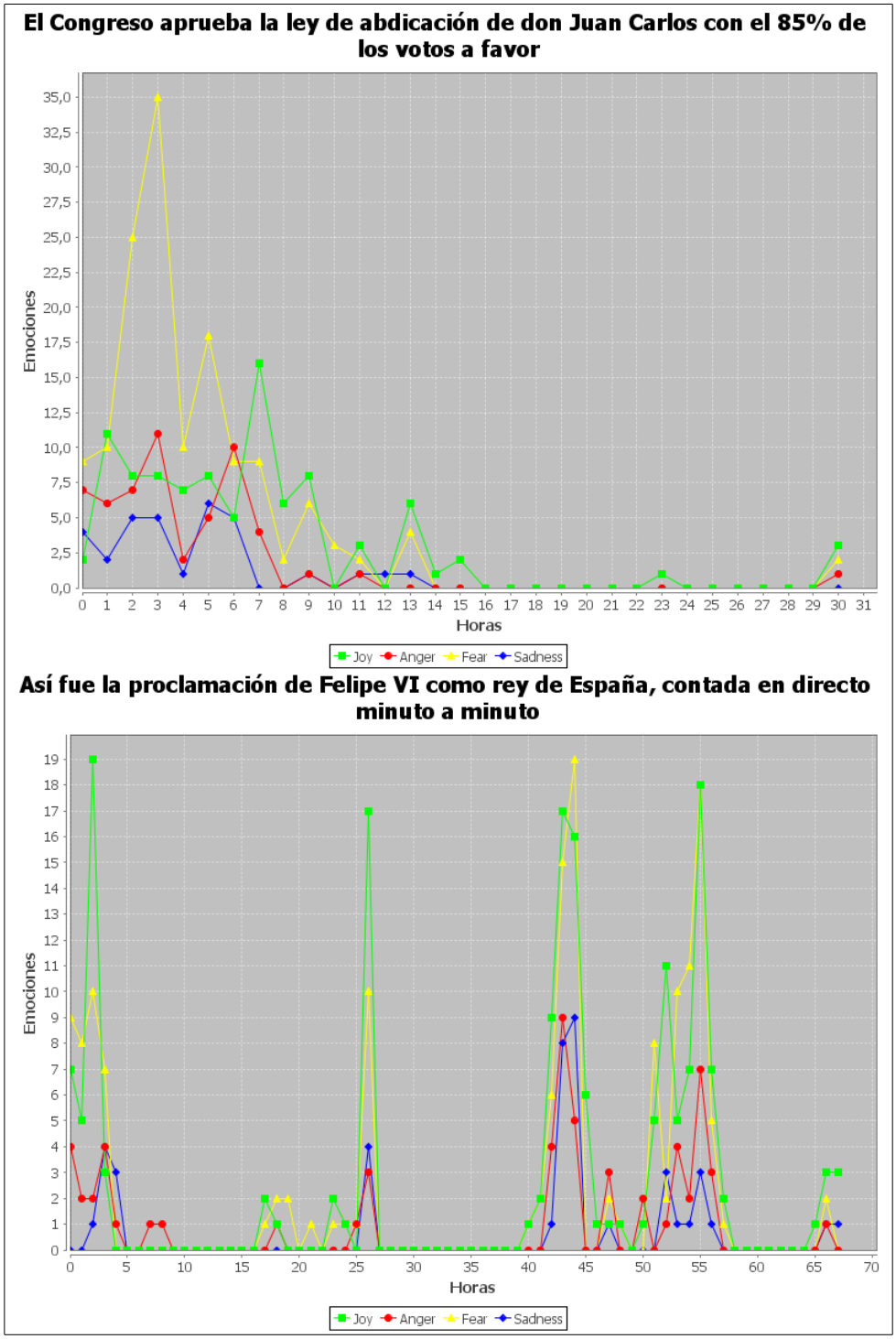

Figura 35. Diagrama de líneas de emociones de noticias relacionadas. Parte 2.

En los diagramas anteriores se pueden observar las emociones detectadas a lo largo del tiempo. Por lo que se observa, parece ser que las emociones encontradas en cada intervalo de tiempo suelen tener la misma proporción entre ellas, aumentando y disminuyendo conjuntamente. La tercera noticia destaca por no seguir este orden, en la que se disparan los valores de las emociones de miedo mientras que los otros valores permanecen más o menos estables.

7.Pruebas y resultados

7.1.Pruebas funcionales

Se ha establecido un plan de pruebas utilizando técnicas de caja negra para detectar el mayor número de errores posible.

Descripción de la prueba	Resultado esperado	Resultado obtenido
Se intenta leer un fichero inexistente para obtener datos previos.	Se muestra un error indicando el problema y se termina la ejecución.	Correcto
Se intenta leer un fichero incorrecto para obtener datos previos.	Se muestra un error indicando el problema y se termina la ejecución.	Correcto
Se lee un fichero correcto para obtener datos previos.	Se leen y almacenan los datos en el sistema.	Correcto
Se intenta agregar al sistema una noticia con una URL incorrecta	No se crea la noticia. Se muestra un error indicando el problema, y se continúa con la ejecución.	Correcto
Se intenta agregar al sistema noticias a través de una URL de RSS incorrecta.	No se crea ninguna noticia. Se muestra un error indicando el problema, y se continúa con la ejecución.	Correcto
Se intenta agregar al sistema noticias a través de una URL del periódico incorrecta.	No se crea ninguna noticia. Se muestra un error indicando el problema, y se continúa con la ejecución.	Correcto
Se intenta agregar una noticia que ya existía previamente.	No se crea la noticia. Se muestra un error indicando el problema, y se continúa con la ejecución.	Correcto
Se obtienen los datos de una noticia compatible almacenada en el sistema.	Todos los campos de datos son rellenados con los datos de la noticia.	Correcto
Se intentan extraer los datos de una noticia no compatible almacenada en el sistema.	No se obtiene ningún dato. Se muestra un error indicando el problema, se sigue con la ejecución y posteriormente se borra del sistema.	Correcto
Se obtienen los comentarios de una noticia nueva	Los comentarios son obtenidos y almacenados correctamente	Correcto
Se intentan obtener los comentarios de una noticia nueva sin comentarios.	No se obtiene ningún comentario y se continúa con la ejecución normal.	Correcto

Se intentan actualizar los comentarios de una noticia existente anteriormente sin haber cambios.	No se añade ningún comentario nuevo.	Correcto
Se intentan actualizar los comentarios de una noticia existente anteriormente con comentarios nuevos.	Se añaden los nuevos comentarios.	Correcto
Se intentan obtener comentarios de una noticia con una URL incorrecta	No se añade ningún comentario. Se muestra un error indicando el problema y se continúa la ejecución.	Correcto
Se guardan todos los datos en un fichero nuevo.	Los datos son guardados correctamente.	Correcto
Se intentan guardar todos los datos en un fichero que ya existe.	El fichero anterior es sobrescrito con los nuevos datos.	Correcto
Se lee un fichero correcto para analizar.	Los datos son leídos y almacenados correctamente.	Correcto
Se lee un fichero incompatible para analizar.	Se muestra un error indicando el problema y se termina la ejecución.	Correcto
Se intenta leer un fichero que no existe para analizar.	Se muestra un error indicando el problema y se termina la ejecución.	Correcto
Se lee un diccionario con nombre y ruta correcta.	Se obtienen los datos del diccionario correctamente.	Correcto
Se lee un diccionario con un nombre incorrecto.	Se muestra un error indicando el problema y se termina la ejecución.	Correcto
Se lee un diccionario en una ruta que no existe.	Se muestra un error indicando el problema y se termina la ejecución.	Correcto
Se analiza un texto que contiene palabras de los diccionarios de emociones.	Se obtienen las palabras detectadas.	Correcto
Se analiza un texto que no contiene palabras de los diccionarios de emociones.	No se obtiene ninguna palabra. Continúa la ejecución.	Correcto
Se intenta crear un diagrama a partir de una noticia con datos de emociones.	El diagrama se genera con los datos esperados.	Correcto
Se intenta crear un diagrama a partir de una noticia sin datos de emociones.	El diagrama se genera sin datos.	Correcto

Tabla 4. Pruebas funcionales

7.2. Pruebas no funcionales

7.2.1. Pruebas de mantenibilidad

En el código de la aplicación se encuentra comentada la descripción de todas las acciones realizadas. Además el software desarrollado se ha diseñado de manera modular. De esta forma se podrán corregir fallos o incorporar nuevas funcionalidades en un futuro de manera sencilla.

7.2.2. Pruebas de fiabilidad

Durante las pruebas funcionales se dieron varias situaciones en las que se producía un error. En todos los casos se muestra a través de la consola de Java indicando el origen del fallo. Si se trata de un fallo que imposibilita continuar el programa se terminará su ejecución.

A continuación se muestra el mensaje que produce el intento de lectura del fichero con los datos extraídos para la aplicación de análisis cuando no se encuentra el fichero.

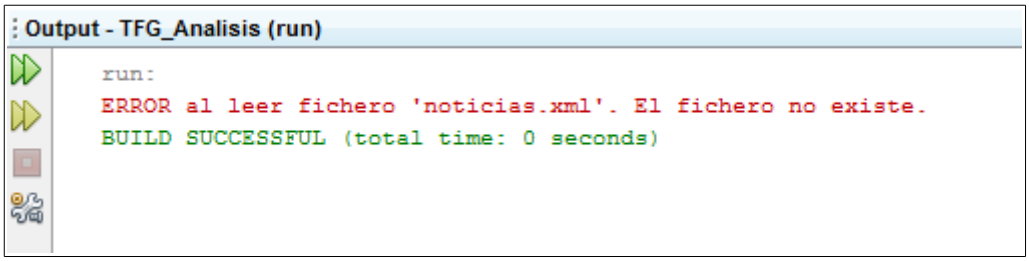

Figura 36. Error de lectura.

7.2.3. Pruebas de rendimiento

Durante la ejecución de las aplicaciones existen varios puntos en los que puede variar el tiempo de ejecución.

En la aplicación de extracción, la mayor cantidad de tiempo se utiliza en obtener las noticias y comentarios. Para una extracción de tres mil comentarios aproximadamente se ha tardado un minuto y veintidós segundos, siendo éste un tiempo muy razonable.

En la aplicación de análisis la mayor cantidad de tiempo se utilizaba en la lectura del fichero con los datos de las noticias. El motivo era el modo de lectura del fichero el cual se leía línea a línea. Como la herramienta XStream necesita todo el fichero, sin necesidad de leer cada línea, se pasó a leer todos los datos del fichero a la vez. Esto disminuyó notablemente el tiempo de lectura para ficheros de 10MB, pasando de aproximadamente diez minutos a diez segundos. Este ha sido el principal motivo para decidir el almacenaje de toda la información relacionada en un único fichero.

El tiempo total de lectura y análisis de aproximadamente tres mil comentarios es de veintiocho segundos, por lo que el tiempo de ejecución se considera muy bueno.

8.Conclusiones

8.1.Conclusiones

En este trabajo se han desarrollado con éxito dos aplicaciones capaces de obtener y analizar emocionalmente noticias de un periódico online.

Los objetivos descritos en el apartado 1.2 de este documento se han cumplido satisfactoriamente:

- Se ha logrado obtener noticias automáticamente de un periódico online de tres maneras diferentes.
- Se consigue extraer la información de cada noticia, incluyendo el texto de la noticia y sus comentarios.
- La información obtenida se almacena permanentemente en disco para un posterior uso.
- Se recupera la información almacenada previamente.
- Se consigue realizar un análisis emocional de los textos de las noticias y de los comentarios.
- Las emociones analizadas se representan mediante diagramas.

Además de satisfacer los objetivos marcados inicialmente, la solución dada al problema propuesto ofrece las siguientes posibilidades para el análisis de las emociones transmitidas:

- Se puede saber el porcentaje de cada emoción, para saber si una emoción o varias son mayoritarias.
- Permite conocer la evolución del número de comentarios a lo largo del tiempo.
- Permite conocer la evolución de las emociones detectadas a lo largo del tiempo.
- Se puede observar el grado de impacto de una noticia viendo la curva de comentarios a lo largo del tiempo. Por ejemplo, se puede saber si tiene una subida constante o por el contrario tiene una fuerte entrada y menor repercusión al final.
- En una noticia que sufra actualizaciones (acontecimientos) se pueden detectar varios picos de actividad en los comentarios. En la gráfica se pueden observar las emociones en cada uno de esos picos de actividad, que pueden ser diferentes debido a estos acontecimientos.

Algunas observaciones tras el análisis de diversas noticias son las siguientes:

- En los textos de las noticias no se suelen encontrar muchas palabras que transmitan emociones debido a que el periodista suele intentar contar los hechos como son. Sin embargo, sí se puede analizar la emoción que esta noticia provoca en los usuarios, mediante el análisis de los comentarios.

- También en los textos de las noticias de longitud corta se detectan pocas emociones. Sin embargo, una vez más, sí se puede tratar de ver qué impacto provoca la noticia en los usuarios.

En este trabajo se han encontrado una serie de limitaciones o posibilidades de mejora que se han añadido como trabajo futuro ya que la realización de este proyecto no los tiene en cuenta como objetivos:

- Los verbos irregulares que expresan una emoción no pueden ser detectados a excepción de su forma en infinitivo.
- La precisión de los resultados de detección de emociones sería mayor si se combinara el análisis realizado con técnicas para el análisis sintáctico de las oraciones, para detectar el sentido de las palabras en la oración. No se ha abordado esta posibilidad por considerarse que excedía del alcance de este trabajo, pero no se descarta abordarla para mejorar la aplicación desarrollada.

8.2. Trabajo futuro

A partir de este proyecto se podrían añadir algunas mejoras para adaptar el proyecto a otras situaciones.

- Una de las mejoras más interesantes que pudiera tener la aplicación sería poder obtener noticias de otros periódicos online. Con ello se podrían obtener los datos de una misma noticia en distintos periódicos y comparar las emociones plasmadas en la redacción de la misma noticia en distintos medios. Como cada periódico publica según una línea editorial marcada por lo que es posible que las emociones detectadas en uno y otro sean diferentes. También se podría comprobar si la influencia que puede tener sobre esa misma noticia tiende a ser la misma en los distintos medios o difiere, o si los usuarios de cada uno de los medios tienden a plasmar ciertas emociones independientemente de la noticia o no. Para añadir nuevos periódicos se necesitaría añadir nuevos métodos de obtención de noticias adecuados para otros periódicos y para la extracción de la información de esas noticias. Una vez guardada la información correspondiente se procedería a realizar el análisis de la misma manera que se lleva a cabo actualmente, al mantener la estructura de datos igual.
- Otra manera de obtener textos para el análisis emocional podría ser a través de foros de opinión al poderse seleccionar un tema en un hilo concreto, blogs pudiendo obtener los comentarios de una entrada específica o Twitter obteniendo todos los “tweets” de un determinado “hashtag”.
- Para aumentar el número de emociones detectadas se podría aumentar el tamaño de los diccionarios con nuevas palabras. Para ello sería necesario que las nuevas palabras expresaran siempre una determinada emoción para que se pudieran considerar válidas. Habría que prestar especial atención a la garantía de validez de dichos diccionarios. Si se considerara adecuado extenderlos, las nuevas palabrasse podrían añadir directamente en una nueva línea en los diccionarios de palabras y el analizador la utilizaría sin necesidad de ser modificado.

- También se podrían agregar distintas emociones a detectar simplemente añadiendo nuevos diccionarios de palabras que transmitan esa nueva emoción. El analizador actual se encargaría de comparar las nuevas palabras con las palabras del texto.
- Los verbos irregulares que expresan emociones constituyen un problema a la hora de detectarlos en los textos, puesto que no se pueden sacar todas sus formas verbales a partir de su forma en infinitivo. Una posible mejora sería la creación de un nuevo diccionario con todas las formas verbales de cada verbo irregular que exprese una emoción determinada. Para detectar estas palabras en el texto bastaría con compararlas una a una, sin añadir ninguna terminación.
- El analizador podría ser mejorado para realizar un análisis sintáctico de las oraciones e indicar si la palabra detectada como una emoción en realidad pudiera expresar la emoción contraria. Por ejemplo, la palabra "preocupación" se encuentra en el diccionario de palabras de emociones tristes. Si un usuario escribe la frase "Una preocupación menos" se utiliza esa misma palabra pero la emoción en la frase es más bien de alegría que tristeza. En este sentido habría que detectar qué palabras hacen cambiar la emoción.
- Con los diagramas de las emociones detectadas en los comentarios de las noticias se puede intuir si una noticia provoca una determinada emoción en los usuarios. Dependiendo del porcentaje de emociones detectadas se puede establecer un sistema de etiquetas para las noticias que permitan saber las emociones que están transmitiendo de un vistazo.
- Utilizando el sistema de etiquetas del punto anterior se podría realizar un filtro de noticias según las emociones que transmitan. Se podría emplear por ejemplo para mostrar en la página principal del periódico sólo las noticias que hayan transmitido alegría o no mostrar las que hayan transmitido tristeza. También se podría utilizar para crear canales de noticias RSS divididas por emoción.
- Se podría realizar un sistema de recomendaciones de noticias basado en las emociones detectadas de los comentarios de los usuarios del periódico online. Si un usuario siempre realiza comentarios en los que se detecten emociones tristes o de ira se pueden sugerir noticias que transmitan alegría.
- Dado que cada periódico suele transmitir las noticias de una manera que se ajuste a su línea editorial habría que comprobar los textos de una misma noticia en otros periódicos para intentar averiguar si intentan transmitir unas u otras emociones. De la misma manera, los lectores de los periódicos suelen elegirlos por su línea editorial. También se podrían analizar sus comentarios para compararlos con los de otros periódicos.
- En cada noticia del periódico 20minutos.es hay información estadística sobre el nivel de actividad de la noticia utilizando los comentarios y las redes sociales. A este gráfico se podrían añadir el número de comentarios y emociones detectadas a lo largo del tiempo. También se podrían incluir el porcentaje de cada emoción de la noticia y entre el total de los comentarios.

8.3. Consideraciones finales

Este trabajo me ha permitido aprender en muchos sentidos. Primeramente, me ha brindado la posibilidad de afrontar un reto real de una dimensión mayor a las prácticas realizadas durante mis

estudios. He podido llevar a cabo un proyecto completo, desde la idea inicial hasta el desarrollo del mismo. He utilizado los conocimientos adquiridos durante la carrera y también he tenido que aprender otras tecnologías con las que no estaba familiarizado. La temática me ha resultado muy interesante y termino este proyecto con ganas de abordar las propuestas de trabajo futuro.

9.Referencias

[1] Catalá, N. & Castell, N. *Construcción automática de diccionarios de patrones de extracción de información*. Dept. Llenguatges i Sistemes Informàtics. Barcelona.

[2] Santana, O., Rodríguez, G. & Hernández, Z. (2003). DAWeb: Un descargador y analizador morfológico de páginas web. *Procesamiento de Lenguaje Natural*, 30, 75-87.

[3] Graña, J. (2002). *Técnicas de Análisis Sintáctico Robusto para la etiquetación del Lenguaje Natural*. Departamento de Computación. Universidad de La Coruña.

[4] 20minutos.es. (2010) *ECO*. Recuperado de http://www.20minutos.es/eco-reputacion-y-comentarios-en-las-redes-sociales/

[5] Rodríguez, P., Ortigosa, A. & Carro, R. (2014) Detecting and making use of emotions to enhance student motivation in e-learning environments. *Int. J. Continuing Engineering Education and Life-Long Learning, 24(2)*, 168-183.

[6] Asociación para la Investigación de Medios de Comunicación. (2013). *Estudio General de Medios*. Recuperado de http://www.aimc.es/spip.php?action=acceder_document&arg=2440&cle=647aff3db0be47080b6fa091fe6aebf672ab7979&file=pdf%2Fresumegm313.pdf

[7] LangPop.com. (2013). *Programming Language Popularity.* Recuperado de http://langpop.com/

[8] Gramaticas.net (2014). *Listado Completo de Sufijos en Español*. Recuperado de http://www.gramaticas.net/2011/02/listado-completo-de-sufijos-en-espanol.html